

SINCLAIR

Every month £1.75 July 1989

QI

4 to 7  
8 11 14 CIRC MOD  
18 22  
30 34  
38 41 42  
46 51

OUTER RECOVER

WORLD

## MICE ART/ THE PAINTER

Two bytes at  
the cherry

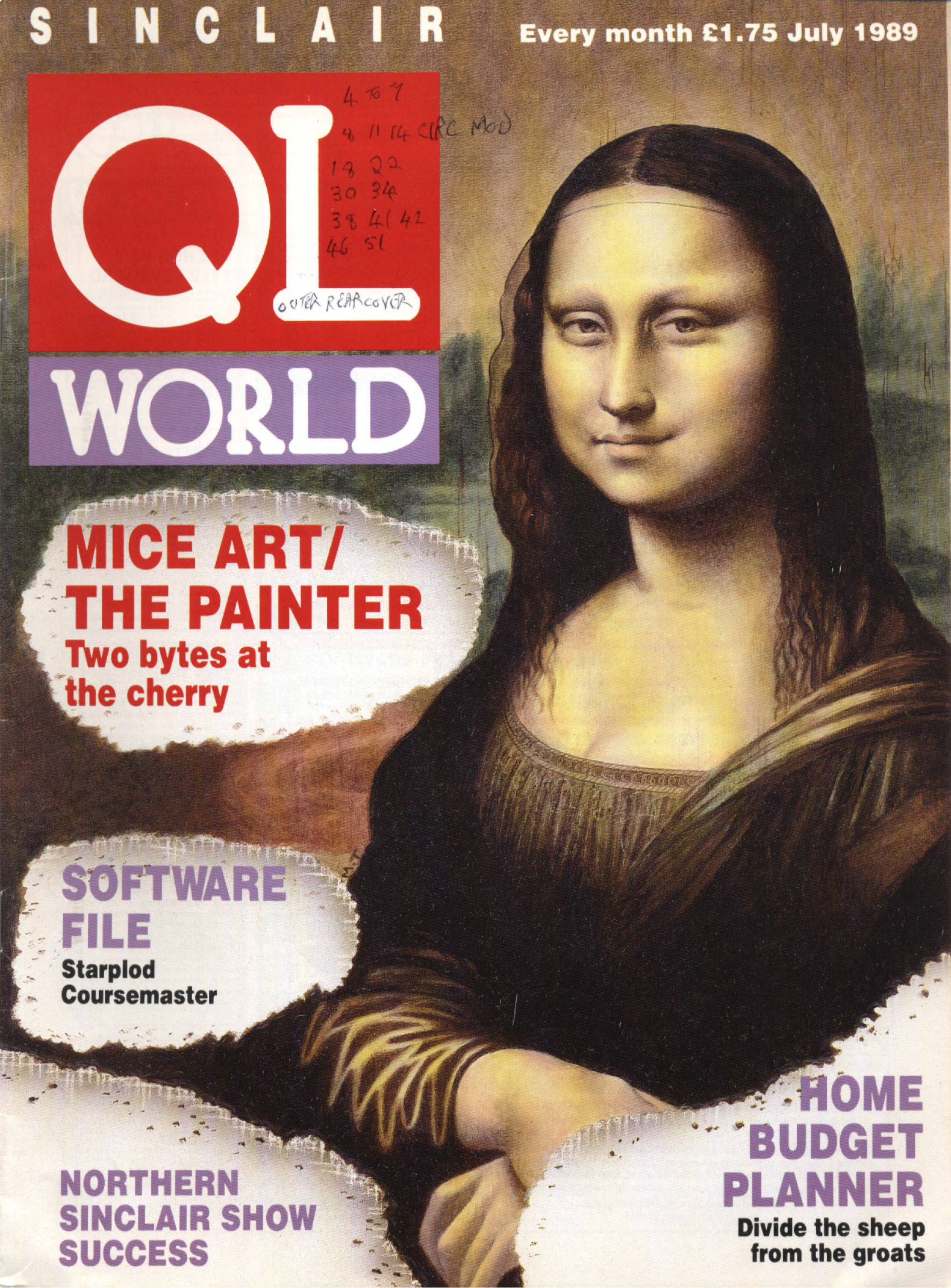
## SOFTWARE FILE

Starplod  
Coursemaster

NORTHERN  
SINCLAIR SHOW  
SUCCESS

## HOME BUDGET PLANNER

Divide the sheep  
from the goats





SINCLAIR



*Editor*  
Helen Armstrong

*Chief Sub Editor*  
Harold Mayes MBE

*Production Manager*  
Nick Fry

*Designer*  
Chris Winch

*Advertising Sales*  
Paul Cave

*Magazine Services*  
Sheila Baker

*Advertising Production*  
Michelle Evans

*Publisher*  
Perry Trevers

*Publishing and  
Commercial Director*  
Paul Coster

*Financial Director*  
Brendan McGrath

*Chief Executive*  
Richard Hease

*Microdrive Exchange*  
089 283 4783/2952  
(2 lines)

Sinclair QL World  
Greencoat House  
Francis Street  
London SW1 1DG  
Telephone 01-834 1717  
Fax 01-828 0270  
Telex 9419564 FOCUS G  
ISSN 026806X

Unfortunately, we are no longer able to answer enquiries made by telephone. If you have any comments or difficulties, please write to The Editor, Open Channel, Trouble Shooter, or Psion Solutions. We will do our best to deal with your problem in the magazine, though we cannot guarantee individual replies. Back issues are available from the publisher, price £2 U.K., £2.75 Europe. Overseas rates on request. Please telephone 089 283 4783 to check availability. Published by Focus Magazines Ltd., London. Distributed by S M Distribution, Streatham, London SW1.

Subscriptions available from: TIL, PO Box 74, Paddock Wood, Tonbridge, Kent, TN12 6DW. 1 year's subscription: £15 U.K. £30 surface mail — Europe and the rest of the world. Add £5 for air mail plus £10 overseas. Tel: 089 283 4783.

Typesetting by Adtec Typographics, Stanford-le-Hope, Essex. Tel: 0375 360967. Printing by Southernprint Ltd. © COPYRIGHT SINCLAIR QL WORLD — 1989

# CONTENTS

■ ■ JULY 1989

- 8 OPEN CHANNEL ● Instruction destruction
- 9 SUBSCRIPTION INFORMATION
- 11 QL SCENE ● Tape search continues
- 12 SOFTWARE FILE ● MacSporran's Lament
- 13 SOFTWARE FILE ● Starplod
- 14 TROUBLE SHOOTER ● Funny you should say that . . .
- 18 PSION SOLUTIONS ● More Archival advice
- 22 THE NETWORKED OFFICE ● Networking is easy
- 26 SOFTWARE FILE ● Coursemaster
- 30 OF MICE AND ART ● A rising star and an old master
- 34 HOME BUDGET PLANNER ● Abacus application
- 37 NORTHERN SINCLAIR SHOW ● New life in Manchester
- 38 DIY TOOLKIT ● MEM device driver
- 41 HARDWARE FILE ● Anti-glare monitor screens
- 42 DIY ARCHIVING ● Housekeeping procedures
- 46 SUPERBASIC ● Making the Calculator user-friendly
- 49 PROGRAM OF THE MONTH ● Cubic colour game
- 51 PROG ● Print Spooler
- 54 PROG ● Multiplication Tables update
- 56 MICRODRIVE EXCHANGE ● Little things to buy

## NEXT MONTH

### PROFESSIONAL PUBLISHER

Digital Precision's desktop publishing package has been substantially rewritten.

### WHAT DID THE INSTRUCTIONS SAY?

Starting with text'87, we look at ways to get more out of popular programs.



# OPEN CHANNEL

Open Channel is where you have the opportunity to voice your opinions in *Sinclair QL World*. Whether you want to ask for help with a technical problem, provide somebody

with the answer, or just sound off about something which bothers you, write to: Open Channel, Sinclair QL World, Greencoat House, Francis Street, London SW1 1DG.

## Not in the book

The following facility with *Easel V2.3* is not mentioned in the User Guide. While in data entry mode, pressing SHIFT F4 brings up the prompt "Enter backdrop colour number". On supplying a number, the graph is re-drawn with an attractive coloured border; all of the standard QL stipples are available. Text and labels are printed on a contrasting strip but axis figures are not, so will disappear if they are the same colour as the backdrop.

Hilary Snaden,  
Portishead,  
Bristol.

## Not by the book

I have for a number of years felt that handbooks for electronics equipment are woefully inadequate. A few weeks ago I advertised for second-hand disc drive, together with an interface and memory expansion and was offered just what I had requested at a reasonable price.

When I received the equipment I found that there was a handbook for the Sandy Super-Qboard and when I assembled the items in accordance with the instructions I found that the various commands for using the drive and for setting up a

RAMdisc worked. The handbook, however, implied that there was a large batch of SuperBasic extensions available but nothing I tried accessed these commands.

Eventually I decided I had been sent a handbook to cover just the facilities I had purchased and that the extensions were not present. Later, in conversation with a more experienced acquaintance, I learned that before these extensions became functional one must ENTER TK2\_EXT.

How stupid of me not to have worked this out myself.

A further examination of the handbook revealed the following paragraphs: "TK2\_EXT enforces the Toolkit II definitions of common commands and functions," and "TK2\_EXT enforces the Toolkit II definitions of common commands and functions. If, for any reason, some of the Toolkit II extensions have been re-defined, TK2\_EXT (c.f. FLP\_EXT floppy disc extensions, EXP\_EXT expansion unit extensions) will reassert the Toolkit II definitions."

This is not exactly calculated to jump out as an instruction to use this command to enable the whole set of extensions.

At several points in the handbook there are sections entitled Beginners Start Here. If this handbook caters for what it regards as beginners, how does one learn enough to become a mere beginner?

Experts should acknowledge that the bulk of their customers expect to use their purchases without necessarily understanding them.

L. Atkins,  
Biggleswade,  
Beds.

Editor's comment: Badly-organised, unindexed, un-

cross-referenced, incomplete, poorly-written and jargon-ridden manuals have been, and remain, one of the plagues of computing. This springs partly from the same attitude which allows poorly-structured and incomplete software on to the market so that the customer can act as an unpaid beta-tester prior to updating.

Early versions are then sold as 'budget' versions with pressure to upgrade when the software proves inadequate or difficult to use and telephone-intensive help-line, or expensive "training courses" offered as a substitute for proper documentation.

I do not wish to under-play the very real difficulties of developing a complex piece of utility software but although computing is still a frontier industry, it is time suppliers were put under some of the same pressure which, for instance, car manufacturers face. Bad documentation, which can render a good piece of software a liability to its owner, results from thoughtlessness on the part of the developers and cost-cutting in refusing to hire a technical author with sole responsibility for the manual. The documentation is produced, often as an afterthought, by technical staff who have little grasp of the layman's needs or language. Apart from the distress to users, doubly to beginners, this factor can and does put businesses into serious financial difficulties.

Despite the gnashing of teeth, QL users have escaped lightly; they have access to an inexpensive machine and comparatively inexpensive software with broad capabilities once they become familiar with the quirks; a number of suppliers provide very good back-up and a serious attempt for the most part to provide adequate documentation.

## Editor's notebook

We have plenty of reviews this month, some of them covering good new adventures which have been a pleasure to their reviewers. There is a dearth of new arcade games but there are still good, older games with which newer users are not familiar. We would be willing to run new reviews of older games. So long as they are still on the market. That is partly why we have included *MICEart* so long after its release. It has never been reviewed properly, and it has given us a great deal of fun.

Not all programs are fun to use — some are a disaster. Some are good; some fail to provide what the user expected, even after serious attempts to establish that. Most software somewhere has a user disappointed or angry through no fault of his own.

I can only repeat that it is unwise to order expensive software, from any source, until it has been reviewed by *QL World* or by one of the user groups, or you or a friend have had a chance to test it. Demonstrations and advertising can only reflect the supplier's view of the goods which, our mailbag indicates, is not always the user's view. The more independent experience you can get the more likely you are to be satisfied with the results.



## More printers from HR5 supplier

QL World advertiser EEC Ltd, featured as a supplier of the HR5 printer in the April, 1989 edition, has contacted us to say that it also supplies two inexpensive dot matrix printers, the GP100 and the GP550.

Any existing user of either of those printers who wishes to write a "printer report" for QL World should write to the Editor, EEC Ltd, 18-21 Miskin House, Chiltern Hill, Chalfont St Peter LS9 9EU. Tel: 0753 888866.

## Microcassettes: the search for tape

Following last month's news that microcassette manufacturer Ablex is faced with a tape shortage which may thwart its plan to lay down stocks of the cartridge prior to ceasing manufacture at the end of 1989, Sinclair Research has responded to QL World concern that another manufacturer should be able to take over production if economics allowed.

Ernie Watkins of Sinclair Research to QL World:

"In the event of Ablex ceasing manufacture of the cartridge, we would be most happy to discuss, with any

interested parties, the possibility of manufacture being taken over. We feel, however, that even with present supply difficulties removed, the practicalities, costs and risks associated with re-locating and re-starting manufacture for a diminishing market may prove unattractive.

"In respect of Ablex considering, because of reduced demand, ceasing production during 1989, this would have been on a planned basis which included laying down reasonable stocks of finished product. At this point any plans or intentions in respect of the

cartridge are frustrated by serious problems on the supply of suitable tape and Ablex is currently awaiting further samples from Germany of assessment."

Ablex production manager David MacSorley had expressed doubt that another manufacturer would resume production, apparently on the grounds that no other company had the tooling capability or experience. Ablex has produced microcassettes for the QL since its development. The current situation regarding tape supply is as yet unresolved.

## Oxfam appeal

Oxfam is launching an appeal for unwanted computer software donations through its London branches. The appeal is aimed at users and dealers and includes hardware, although consumer software and games are the most popular donations. The software will be resold through London and larger Oxfam shops as part of a special promotion to raise money for long-term and emergency famine relief.

Richard English at Oxfam says it hopes to sell software for some makes of computer exclusively at specific branches, so that users would know where to look.

Donations can be accepted at any Oxfam shop branch country-wide during June and July. The resale promotion begins on June 27.

## Fractals grow

The specialist fractals newsletter *Fractal Report* has now reached issue 1, following issue 0 — the free introductory issue — and issue -1 — the flyer. Issue 2 is due on September 1.

Fractal Report, published by Reeves Telecommunications Laboratories Ltd of Truro, contains articles and programs concerning fractals and "similar iterations in one or more dimensions". Authors retain copyright of their material. Issue 0 has 23 pages on fractals, Mandelbrot patterns, computer graphics and mathematics, A4 with a yellow cover.

Fractal Report now has 100 subscribers. Issue 1, out at the end of April, contains articles on speeding calculations, exploring the Mandelbrot set, and others.

The U.K. subscription costs £10 for six issues, backdated to the start of the current volume. European subscriptions cost £12, printed paper rate; elsewhere costs £13 or \$23 — printed paper airmail rate. Enquiries and subscriptions to RTL, West Towan House, Porthtowan, Cornwall TR4 8AX. Send an A4 or A5 SAE for information.

## APOLOGIES

Inside the June issue of QL World an advertisement placed by Digital Precision drew a comparison between an MS-DOS emulator program by Ant Computing and their own emulator program "The Solution" and we would like to make it clear that QL World and its publishers in no way support or endorse any of the general statements made by Digital Precision regarding other unreviewed emulators and that we wish to apologise to

Schon PCP if any readers of the magazine felt that to be the case.

Schon PCP are presently introducing a new emulator program "The Transformer" and they have informed us that there is no connection between the Ant Computing program and their new emulator and that no evidence has been produced to support any suggestion that the latter will not meet its specification in full.

As with any new software programs we advise our readers to wait for independent reviews.

## Report Writer published

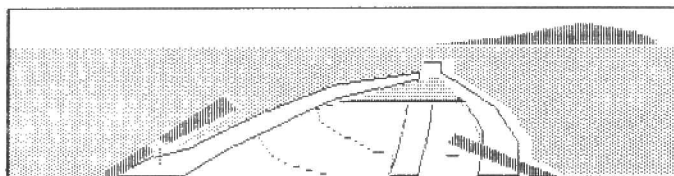
Report Writer is a program designed to assist the writing of reports and critiques. Developed by a teacher to help with pupil reports, it has been adapted to be useful with any type of structured report.

The program is based on two files, a Report bank, carrying the structural material, and a Comment bank, carrying the commentary for updating and editing.

Report Writer is available on 3.5in. disc or Microdrive for £14.99 from Cottage Enterprises, 6 Shorwell Close, Grantham, Lincs NG31 7II, and is accompanied by a printed 15-page instruction booklet.



# SOFTWARE FILE



Here, you can see:-  
A set of bagpipes.  
> d  
The sea-cave beneath the Cellan.  
  
Here, you can see:-  
A piece of driftwood.  
A ladder.  
> enter boat  
You board the boat.  
The boat rocks uneasily on the swell. The oars are by your side. Dare you  
row to Snottay or will you disembark?

**N**ow that the Digital Precision ACT is beginning to be digested by adventure writers, a few very professional adventures are starting to reach the QL scene. Being a reasonably-experienced adventurer I am very pleased to say that this latest release from CGH Services alias QL Adventurers' Forum is certain to be a winner.

It is written by Dave, Ann and Katy Watson who live in Scotland and it contains all the necessary ingredients to keep anyone glued to their sets for many hours.

Being a typical ACT adventure it has all the good points of the original game *Imagine* developed by Steve Sutton using the system, i.e., the location scenario is shown clearly and as you drop or pick up articles they appear or disappear from the screen.

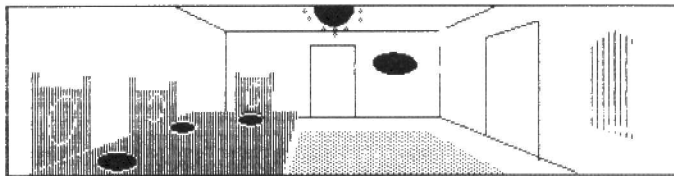
The text of the adventure can be in mode 4 or 8, for TV or monitor by a changing the boot listing. By typing-in the word 'Info' you are given several useful commands, such as "HEALTH, LOOK, SCORE, SAVE, RESTORE, GET, DROP, RESTART, EXAMINE, INVENTORY, READ, DRINK, THROW, OPEN, DIG, QUIT AND ATTACK" and they for the most part are used in the so-called verb-noun category, i.e., 'Examine Stairs' or 'Lift Carpet'.

The direction movement is achieved by typing-in the first letter of the appropriate compass bearing, i.e., (N)orth.

The four-page instruction manual is clear and helpful.

Full details are given of how to make a back-up, the Basic commands, general advice to new adventurers and a very important introduction. It must be read thoroughly as clues are contained in its text.

"... Reckless, seeking only fame, fortune and adventure,



> v  
The dining room.  
  
Here, you can see:-  
A broken bellpull.  
> look  
The vast and draughty dining room is dominated by the long table set for supper. The silver shines coldly. In the chill air the chandelier tinkles above the table. A door leads to the kitchen and another to the Great Hall.  
  
Here, you can see:-  
A broken bellpull in the corner of the room.

you ride the moorland wilds north of the great Glens and snow-mantled peaks of the high Bens. This is Scotland in the year 1745, a wild, savage country, its inhabitants no more hospitable than the peat bogs of rough heather and scree. Clan wages war on clan and a lonely traveller like yourself is fair game to any. . . "

## Bagpipes

From the moment you start the Adventure your attention is grabbed. First you hear the sound of bagpipes playing a sorrowful lament; then the opening scene, drawn in the top third of the screen, shows the silhouette of a grim castle and

beneath the following words appear, written in a strange ancient script:

"The Bonnie Lassie O' Invernoddie has disappeared. A huge reward has been offered for her safe recovery; lured by this and the tales of her legendary beauty, you come at last to grim MacSporran Castle. . . "

"... Then you find yourself outside the Castle with roads leading away but is there any point in taking them, for all routes seem to end inevitably back at the grim castle? As you

the portrait of the 'Laird of Auchtermidden'?"

"Why can I not wear the kilt of the dress MacSporran?"

"Should I play the 'Pipes of Ewan MacCrummock'?"

I confess that seldom have I been so intrigued with such an original adventure, perhaps also the first in Scottish? As each scene is displayed on the screen, I find myself marvelling at the amount of thought and ingenuity the family Watson has put into the script. Instead of the usual bland "I do not understand Lift", when you have entered a word not in the vocabulary this computer replies, "... 'Lift' is a word I dinna ken!" In fact, the whole adventure is beautifully lifted with Scottish phrases and descriptions.

Although humour is present throughout the scenario, I assure you that solving this adventure is not child's play. If anything, when I started, I was deceived by the apparent simplicity of the whole plot. Do not be fooled as I was, for the authors have put much thought into creating an intriguing and absorbing puzzle.

I have visited almost all the 60 or so locations but have not solved all the traps and mysteries, for this is a very deep and thought-provoking adventure. All in all, a first-class piece of writing which will give great pleasure to all those who enjoy "Escaping to far-off Lands"

**John Shaw**  
straps on his  
sporrans and  
heads for the  
highlands.

and produced at a very reasonable price.

I have found no bugs or errors, so it seems that the family Watson has done its homework and produced a very professional package. It will run on 128K or expanded machines.

I have not as yet found the "Bonnie Lassie" but make no mistake, I shall.

## Information:

**Program:** MacSporran's Lament

**Price:** £8 disc or Microdrive

**Supplier:** Services: Cwm

Gwen Hall, Pencader,

Dyfed, Wales SA39 9HA.

Tel: 055934 574.

enter each new location a fresh picture is quickly drawn on your screen, adding greatly to the atmosphere and enjoyment

"Wandering through the rooms, four upstairs, four down, and a Tower, admiring the shields and swords on the stately walls, all seems very quiet; but the silence is deceptive and curious things keep taxing the mind. Ghostly hands touch you and whisk you to underground locations. Secret doors open - only to the correct command - and grim warnings appear when you linger in certain places.

"Why, when I pull the broken Bellpull, do I hear the ghostly lament?"

"What is the significance of



Things are pretty dodgy abroad the Galactic Federation Space-Base. After plodding through the galaxies where no QL has plodded before our ship has picked out a likely solar system for us to try for size. Unfortunately, when we woke from our two-million-year cryogenic sleep we found that critical damage had occurred to the ship during our long flight from Earth.

The Space-Base is dangerously short of moronium, the precious mineral essential to the life-support systems generator. There is only sufficient to last a few days, or at most a week. Naturally, as Commander, I am going to order some poor moron to find more moronium – to put his life on the line for the rest of us 10,000 crew members – but who to send out in the small Gal-Fed exploratory craft? What about that ensign with the long black antique computer...?

Yes, you guessed it. It is you. So begins your mission in Alan Pemberton's new adventure game, *Starplod*.

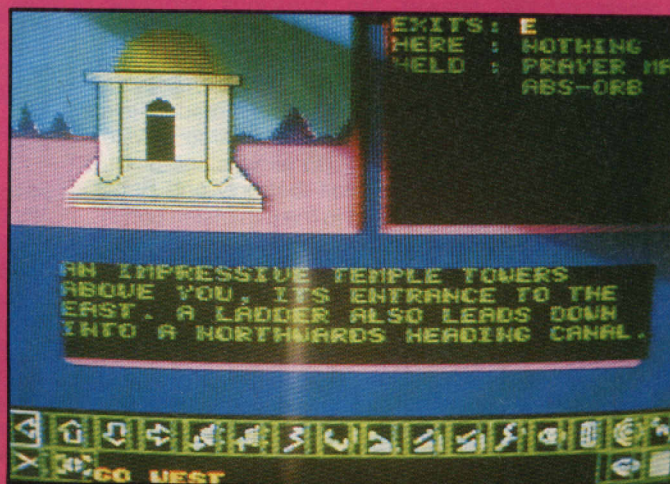
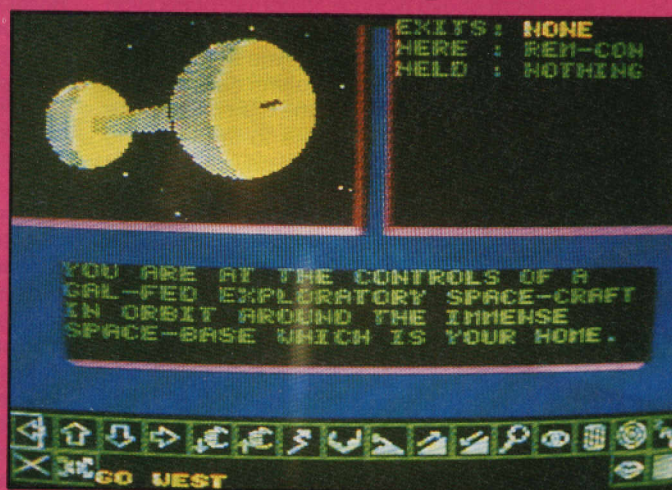
The first thing with which to come to terms in this delightful game is the fact that it is icon-driven. Using either arrow keys/spacebar or joystick, all commands are effected by selecting one of the 20 symbols in the icons/command window. Although this system perhaps reduces the complexity of the game, it has the advantage of ensuring that every command is executable. There is none of the irritation associated with responses from traditional parsers like "I don't understand. Try some different words."

One soon becomes adept at choosing each command. The neat thing is the way in which the object of some commands is highlighted in the text window. FIRE AT allows you to move through the location description until you light on a likely target. In the case of manipulating objects, after the command, e.g., ACTIVATE, the available items are scrolled through one by one until your reach the object you want.

The command THROW takes this method one stage further by requiring first the object to be thrown from the list and then the text window target at which to lob the chosen item. If this sounds

# SOFTWARE FILE

## Hyperdrive and icon-drive come to David Watson's aid in the depth of space.



### Information:

**Program:** Starplod

**Price:** £8 (£6 if you supply media plus p&p), includes manual.

**Source:** CGH Services, Cwm Gwen Hall, Pencader, Dyfed, SA39 9HA. Tel: 055934 574.

**Captain's Log:** Stardate 2000000.5

complicated, do not worry. In practice it is simple and neat.

Above the text window are two more displays. The one on the right gives information on exits from the current location and shows what objects are present and what objects are held. On Hyperdrive it also shows the solar system. To the left is the graphical display of your whereabouts and natty little pictures they are, too. I like particularly the effect when one Hyperdrives to a new location.

So how do you go about getting the required moronium, because you will not be allowed back on board the Space-Base until you have the darned stuff? This particular solar system contains five planets to which you will have to Hyperdrive. Once in orbit round a planet, you transport down to the surface. Is it me, or does the Transpad look suspiciously like a Big Mac?

Be warned. Some planets are inhospitable places. Then it is down to getting hold of the available objects and finding exactly what you are supposed to do with them. The game allows for Save and Ramsave and this is particularly useful because you will almost certainly be zapped at least a few times on your danger-fraught expedition.

Starplod is a charming game, well-thought-out and containing much of the whimsical humour which is the hallmark of Pemberton's work. One of the objects is a prayer mat. Trying to UTILISE it will in most cases return with "There is no Divine response." As a piece of programming, Starplod is very elegant. Perhaps the inclusion of sound might have added another dimension. My only real criticism is that I found it a little easy and the pleasure was over all too soon. At £8 you will have to trek or plod far to find comparable value.



# TROUBLE

Bryan Davies glances at some contenders, fixes Trump Card lock-ups and explains overseas subscriptions.

**T**he Digital Precision MS-DOS emulator was delayed somewhat by problems with MS-DOS. Version 4.00, which supplier Microsoft has apparently been selling, has received several bad reviews in PC magazines and looks to be too full of bugs to be safe to use. Version 4.01 should be out now and this is what DP hopes to ship with the "chocolate" version of its emulator.

The £50 difference in price between this and the "vanilla" version is due to the cost of Microsoft MS-DOS 4.01 which has the full MS documentation — good value, as it seems unlikely that 4.01 will sell for as little as this from the usual PC suppliers.

The vanilla version can be used by those who already have a copy of MS-DOS; the emulator has been tested with a variety of earlier versions, such as 2.10 and 3.30. For users who wish to swap work between QL and PC, the program provides the facility to read/write to DOS-format discs.

The upgraded *Flashback* was targeted for release in March-April. The report generator module follows. The beta test version has some other significant enhancements from the original version. *text<sup>87</sup>* version 2.0 has been released; this is now a mature program which can be used for serious work and incorporates features usually found only in much more expensive programs — or not found at all.

The modification to Trump Card mentioned previously as a potential cure for lock-ups seems to work well. Miracle Systems has been making the modification from some time and reports good results; I have tried it on two Trumps with success. It is simple to do but only for those who are good with a small soldering iron, as it involves putting resistors piggy-back across existing ones.

Do not try it unless you are proficient with the soldering iron, because there are several very fine PCB tracks and the legs

of an integrated circuit in the vicinity; you might join the wrong points and ruin the Card.

At the top right end of the PCB — seen from above when in its normal orientation to the QL — there is a small IC with a row of eight resistors alongside it on the left. The resistors are 560ohms each and are touching each other; the right ends of the resistors are very close to the left-hand row of legs of the IC. If in any doubt about which resistors are meant, do not attempt the modification.

What has to be done is to solder an 820ohm resistor across each of the eight existing 560ohm resistors. The values of the existing resistors are being reduced by putting a similar-sized resistor in parallel with each one. If you do not feel sure about making this modification, contact Miracle Systems direct.

It has been pointed out that using the Toolkit command RJOB 0,0,0 does not "kill" SuperBasic but merely produces the message "not complete", effectively ignoring the command. What I had in mind when writing about RJOB was the related command SPJOB 0,0,0. The latter reduces the priority of SB to 0; with my set-up at least, this appears to make SB inaccessible, because the input line has gone. The only way I know then to get SB back is to re-set the machine.

## Buffering

A reader has asked for details of the necessary connections to enable several devices to be plugged into the 64-pin expansion port. He was hoping to fit several similar connectors in parallel with the existing one but was advised that some form of buffering is required. I could not answer the questions properly; perhaps other readers would write with explanations of what is required?

Does a multi-way expansion device have to be buffered? Another reader says that SPEC sells an unbuffered device which works well. Do all versions of QL suffer from the fault that only one ROM device connected to the expansion port is recognised, rather than the 16 specified?

Are all 16 ROM slots taken up when the Trump Card is fitted and is it then impossible to connect anything other than disc drives? The Miracle hard disc unit connects to the ROM port and a Trump Card will still function as usual in the 64-pin slot when the hard disc is connected.

You may not have noticed that com-

mercial programs are supplied on discs which have the write-protect tab in the "write enable" position; it is good practice to move the tab to the "write protect" position as a first action when the disc is received, in case subsequent actions cause a drive to attempt deletion of files on the disc.

## MCS Interface

**Mort Binstock** writes from the U.S. to offer a solution for those users — e.g., **C. Roger Fernando** — of the MCS interface who have problems using programs such as *Super Media Manager* and *DiscOVER*. The problem in this case is the lack of the direct sector addressing facility in the interface. MCS has a utility routine which adds the facility — it can be put into a boot program. Whether or not this is freely available I do not know but the address to contact is given in the information.

**A. R. Fuller** has had a variety of problems using the ANT MS-DOS emulator version 1.0; if any other readers have bought this software we would appreciate comment on it. A quotation supplied from the brief instructions with the program suggests that the author is inviting users to obtain "pirate" copies of MS-DOS, which is not a practice we can support. Given the size of Microsoft, and the fact that U.S. software companies are jealous about their proprietary rights, it is also asking for trouble. It is understood that a version 2 of the program is being worked on.

Replies have not been received to queries sent to various suppliers in the last few months. Among them are **ABC Elektronik** — re **Cornez Pierre** and **Schoen** — re **Henri Hulet**. Perhaps these suppliers would now like to comment on the problems experienced by those readers?

## Subscriptions

**R. Gilbert** from Nova Scotia asks why the charges for software are higher to overseas buyers than to U.K. ones and quotes figures to suggest they should be lower. I do not think the reasons can be made fully clear to anyone who has not been in the business of sending goods overseas. The fact is that sending goods overseas is much less simple than sending them in the U.K. and the administra-



# SHOOTER

E M S O L V E D

tive cost, largely staff time, is much greater. While you can put a personal airmail letter into a local postbox and it will not cost much more than a local first-class letter, commercial packages have to be taken to a Post Office and forms filled in. If you are VAT-registered it is necessary to have all packages listed on a Post Office form or you are liable to have difficulty with the VAT people, especially if a package is returned, when you can be charged for it.

## Despatch

The reason for some suppliers not deducting anything from the price of software to allow for no VAT being charged is not that they want to pocket the VAT but that they are making an effective increase in the price of the goods to cover some of the extra cost of despatching them.

To suggest, as Gilbert does, that postage charges for sending packages overseas are small is wide of the mark in many cases. To take the extreme, there is no way instruction manuals of the size Digital Precision supplies with its software can be sent for the normal basic postal charge; the cost is likely to be pounds rather than pence.

The reasons for magazines like *QL World* costing so much more overseas are rather different. The print trade has its own agreements round the world and publishers will not supply magazines direct to readers at a price lower than that charged by their agents in the countries concerned. The cover prices of magazines are normally much higher in foreign currency so, inevitably, the subscription costs will be higher, too.

Because there are fewer people concerned with the QL in a country other than in the U.K., sales will be low whatever

price is charged for a magazine and, as with shipping software, the administrative cost will be appreciably higher.

## INFORMATION

**MCS interface:** Micro Control Systems, Electron House, Sandiacre, Nottingham NG10 5BA. Tel: 0602 391201.

**Palantair Products** software is now handled by Rob-Roy Software, 94 Teignmouth Road, Clevedon, Avon BS21 6DR.

**MS-DOS emulator:** Digital Precision, 222 The Avenue, Chingford, London E4 9SE. Tel: 01-527 5493.

**Trump Card:** Miracle Systems, 20 Mowbarton, Yate, Bristol BS17 5NF. Tel: 0454 317772.

## text<sup>88</sup> VERSION 2.00 STATE OF THE ART IN QL SOFTWARE

**New:** German and French versions of all our software with full manuals are now available. Advanced text-mode drivers for the new Star printers have been added to the range.

Previous versions of text<sup>87</sup> have been rated as more advanced than any PC wordprocessor in screen display and printing. The new version 2.00 goes even further by providing:

- ★ Justification on the screen as well as in print.
- ★ Display of the ruler for the cursor line. Display of the name of the typefaces in addition to the WYSIWYG display of typefaces by multiple graphic founts. Ten screen founts are now supplied with text<sup>87</sup>.
- ★ Easier selection of typefaces.
- ★ New 80 page manual.

Advanced features available in earlier versions have been enhanced further. The following are just some of the features.

- ★ Extremely fast. Much faster than speeded-up Quill.
- ★ Completely menu driven. No obscure key-press commands to remember.
- ★ Full configuration from within the program. The configured program automatically loads the appropriate founts and printer driver when executed.
- ★ Trouble-free handling of documents of any size — memory is the only limit.
- ★ Fast dynamic block operations.
- ★ Advanced page layout designer allows header, footer and up to 4 columns of text on the page.
- ★ State of the art handling of daisywheel printers. Supports multiple daisywheels with different pitches including proportional spacing. Supported printers include Epson, Brother, Silver-Reed, Juki, Qume, Quen-Data, Triumph-Adler and many more.
- ★ State of the art handling of 9-pin dot-matrix printers in text mode. Different widths of characters, superscript, subscript, bold, italic, underline can all be mixed in any part of the text. The twenty different printer drivers supplied support more than a hundred different models without any modification. Built-in support for accented characters.

**founttext<sup>88</sup>** The state-of-the art graphic printer driver for text<sup>87</sup>. Now supports many non-Epson printers as well as Epsoms and compatibles. **founttext<sup>88</sup>** is supplied with 32 high-quality founts in different styles and sizes up to 72 pixels high. With **founttext<sup>88</sup>** you can use graphic founts without the limitations in text editing and document size imposed by other desktop publishing programs. Full WYSIWYG. Special 24-pin version at no extra cost.

**fountedit<sup>89</sup>** The graphics editor for **founttext<sup>88</sup>**. Allows you to create founts of up to 84 × 96 pixels. Captured screen images can be loaded to produce picture founts for use within documents. Extensive fount editing operations include copy, import, inverse, reflect, shift, etc.

**2488** State-of-the-art text-mode dedicated printer drivers for Epson, NEC and Star 24-pin printers and compatibles. With these drivers, text<sup>87</sup> is the only QL program that can use the advanced features of 24-pin printers currently available at affordable prices. The drivers support multiple typefaces, proportional spacing, double-height and double-width modes. Epson and Star drivers also support shadow and outline highlights. Drivers support up to 1200 different combination of typeface and highlight.

text <sup>87</sup>	£45	founttext <sup>88</sup>	£25
fountedit <sup>89</sup>	£15	2488	£15
Complete edition: text <sup>87</sup> + founttext <sup>88</sup> + fountedit <sup>89</sup> £80			
German version: add £4 to the total			
Other software			
Taskmaster	£25	Qtyp	£29
Spellbound	£29	Qpac	£19
Flashback	£40		

Personal callers and credit card orders are welcome by **Care Electronics**. Tel: 0923 672102.

Prices are inclusive of airmail worldwide. Payable by cheque, Postal Order, Eurocheque or credit card. Please specify language, cartridge or disk.

text<sup>87</sup> requires memory expansion (as little as 64K will do). founttext<sup>88</sup> and fountedit<sup>89</sup> require at least 128K expansion.

See the reviews in *QL World* (April) or *Quanta* (March). Send for our free leaflet if you need more information.

An independent telephone support service is now available. Contact Mr Terry Harman on 0604 842875 for details of the charges.

**Software<sup>87</sup>**  
33 Savernake Road,  
London NW3 2JU

**text<sup>87</sup>**



## Ron Massey picks up a mixed postbag.

# • PSION

Reaching into the bottom of the mailbag, I discovered some backdated correspondence which, much to my embarrassment, had been neglected. With the possibility that other readers may have had similar problems — and in the hope that the original correspondents have long since had their problems solved — we open this edition of Psion Solutions.

A Psion Solutions reader has asked if there is any way to disable Quill page numbering. Within certain limitations — mainly a maximum starting page number of 254 — Quill page numbering is flexible.

Users are offered the options of Arabic (1, 2, 3), Roman (I, II, III) or alphabet (a, b, c) page numbering. Numbering may be positioned, with optional additional text, in the header, footer, or both. If used, headers and footers will be printed on each page of a document and may, for example, consist of: My doc; Page *nnn* of 100.

Quill offers four options for header and footer positioning — centre, right, none and left. Writing single-page documents which includes superfluous page numbering can be a nuisance but the Quill numbering can be switched off.

Defaulting to no header and a footer consisting of page *nnn*, switching the footer off can be done sequentially by pressing <F3>, <F>, <SPACE> (twice). The footer will be set to NONE. Press <ENTER> and you will be back into your document.

Another reader asks if there is a way to delete forced page breaks. To force a page break, press <F3> <0> <P> <P>. The page break will occur at the line immediately below the cursor position.

To delete a page break, move the cursor to the first line following a page break and then up one line. If the cursor stops on the line it is a forced page break and can be deleted by pressing CTRL and the left cursor key.

A fellow Canon PW1080 printer enthusiast has written with two rather odd problems. His Canon is fitted with an internal serial interface; he does not say whether it was supplied by Canon or is of third-party manufacture. The version of printer software was not specified.

The first problem occurs with printing Quill documents. After printing about one-third of a page of text, the printer prints "Overrun error" and scrambles the remaining text.

The second problem, this time with Easel, prevents a complete dump being made without reducing the dump to near unintelligibility. He adds that the "space invader rubbish" is delayed depending whether or not DIP switch 2-3 is on or off. This DIP switch defines whether the 3K input buffer is set as an additional character buffer or as a PCG.

The problem described could possibly be attributable to either or both of two sources. The first and most suspect may be the interface an may be indicating timing or baud rate problems.

I had a similar problem with my external serial-to-parallel interface some time ago which would infrequently and unpredictably drop characters from a line of text. It also had the annoying habit of adding characters spontaneously to a graphics dump. Changing the interface cured the problem.

### Graphics

With regard to the second possibility, try altering the settings of SW3-1 and SW3-4 to their original factory settings of ON and OFF respectively. SW3-1 sets the Canon to internal fixed Select mode, when on. SW3-4 selects either automatic linefeed, when on, or a linefeed on a linefeed signal only, when off. If that does not work, try setting the two switches in each of their four possible permutations:

SW3-6W3-4  
OFF OFF  
ON OFF (factory setting)  
OFF ON

Another printer problem; this reader wants to convert GPRINT\_\_prt so that it will print through ser2. Modifying any binary file is simple but always requires that you know the exact position in the file what you want to alter is located. My GPRINT\_\_prt file is 510 bytes long. I mention this only because other versions of the graphics dump may have been supplied by Psion.

Since the graphics printer dumps are not supplied with version numbers this may be the easiest but not most accurate way to find if your GPRINT is the same as mine. Before you start, copy GPRINT\_\_prt to the same drive — or a RAM disc — under another name. COPY mdv1\_\_GPRINT\_\_prt to mdv1\_\_GPRINT will work for the purpose.

There are two ways to modify machine code files. If you have *The Editor*, load GPRINT\_\_prt with the RU command. The sole occurrence of SER1 appears on line two, column 81. Change the "1" to a "2" and write the file back to a drive. I hasten to add you should do this with a back-up copy only.

The second method works just as well but relies on your GPRINT\_\_prt being the same as mine. In sequence, type-in the following commands in SuperBasic:

```
a=RESPR(600);
LBYTES mdv1__GPRINT__
prt,a
DELETE
mdv1__GPRINT__prt
(Do the latter with a backup
only).
```

This will load the file into a specified area of memory. To modify it, type:

```
POKE a+273,50
SBYTES mdv1__GPRINT
__prt,a,510
```

To see if your modification has been successful, type: COPY drive\_\_GPRINT\_\_prt to scr  
SER should appear on-screen in line five, column

```
100 MODE 4 : CLS
110 REMark IF You have a
120 REMark disk interfact
130 FORMAT RAMS_3
140 COPY dv$&"printer_dat
150 PRINT "1. Quill"
160 PRINT "2. Archive"
170 PRINT "3. Abacus"
180 PRINT "4. Easel"
190 REPEAT main_lp
200 key=CODE(INKEY$)
210 SELECT ON key
220   = 49 : Prep_psi : o
230   = 50 : Prep_psi : o
240   = 51 : Prep_psi : o
250   = 52 : Prep_psi : o
260 END SELECT : END REPE
270 EXEC_W "flp1_" & prog
280 STOP
290 :
300 DEFINE PROCEDURE Prep
310   CLOSE#1 : CLOSE#2
320   WINDOW#0,400,20,35,5
330   OPEN#1,"con" : OPEN
340   END DEFINE
```

### A simple boot for new disc users.

seven and should now be followed by 2. If the "2" appears anywhere else, your dump is different from mine and the easiest way to obtain your file position is to copy the GPRINT\_\_prt file to the screen (scr) again and count the number of characters from the beginning of the file to the character following SER.

Once you have obtained the character position, modify the POKE a+273,50 to POKE a+actual\_\_position,50. Try making a dump from ser2 using the modified routine.

A reader has upgraded his system to disc drives and RAM expansion and wishes to transfer his programs to his new system. There are several ways this can be done, including using the mdv emulation feature included with most new disc systems — i.e., flp\_\_use mdv.

Drive emulation, while useful for some requirements, defeats the purpose of upgrading. The easiest way to make a permanent conversion to a disc system is to spend a little time organising your files. The best way is determined only by



# N·SOLUTIONS·

AM disk on your  
toolkit:

TO RAM8\_printer\_dat

```

pg$="Quill"      : EXIT main_lp
pg$="Archive"    : EXIT main_lp
pg$="Abacus"    : EXIT main_lp
pg$="Easel"     : EXIT main_lp
main_lp

```

si

5

2, "con"

how you prefer to work but here is a starter:

1. Format a disc.
2. Type-in the foregoing listing and save it to the disc with the filename BOOT. If you are using DP *Lightning*, save the above as Boot2 and use the *Lightning* boot-maker utility to make your BOOT. As you add programs to your applications disc you will probably want to add more features to this simple program, such as setting the date and time for date-stamping your files.
3. Using the normal "copy mdv1\_\_filename TO flp1\_\_filename" procedure, copy the following files from cartridge to your newly-formatted disc:  
Quill, Archive, Easel, Abacus and one copy each of Config\_\_bas, Install\_\_dat, Printer\_\_dat, Install\_\_bas and GPRINT\_\_prt  
Optionally, you can copy the following:  
Quill\_\_hob, Archive\_\_hob, Easel\_\_hob and Abacus\_\_hob  
Since the \_\_hob files take up a fair amount of disc

space, copy them only if you use them. If the program does not find the file when you press <F1> it will say so and allow you to continue.

4. After re-setting your QL, run Config\_\_bas.
5. You will be asked for the new default locations of three files:  
System files (Help) — Enter flp1\_\_  
Printer data — printer\_\_dat  
file: Enter RAM8\_\_  
Data files — Drive to save to: Enter flp2\_\_
6. Follow the screen prompts and, after each program in turn has been configured, go back to SuperBasic and type "GOTO 1". This will re-start the process and you can configure the next Psion program.

A Spanish reader would like to know if it is possible to alter the prompts of his U.K. version of Quill to Spanish. He has been advised correctly that Quill cannot be used for the purpose. The answer is, within limits, yes; Quill or any program can easily be modified in this way.

Using Quill as a binary editor will not work. For one thing, Quill displays only the printable characters from the QL font set. For another, attempting to import Quill into Quill via the IMPORT option will load correctly only the first seven bytes.

Making alterations to any binary file — in this case a machine code program — is fairly simple. A great deal of care must be taken not to alter the program and, because you have no accurate indication in the file which prompts are used for which commands, you may have to do some of the modification by trial and error.

A long-winded way of altering the prompts, one which requires a great deal of time and patience, is to write a SuperBasic program which will POKE new values into a memory-resident file. The procedure is outlined for GPRINT\_\_prt. Since Quill is about 57K in length this

method will require a great deal of counting and more than a little luck.

The safest way of altering a binary file of any kind is to load the file into an editor — not Quill. As a rule of thumb you can usually modify any text strings you can read. There are two inflexible rules you can violate only at the risk of your program — ensure that the editor is in over-strike mode as opposed to insert mode — this will help prevent violating rule 2. Under no circumstances try to increase the length of the word you are modifying — doing so will usually destroy the program.

Because of rule 2, any Spanish words which are longer than their English equivalents will have to be abbreviated. Conversely, any words which are shorter can be followed by spaces up to and including the last character of the English word. For that reason I do not recommend making global substitutions with an editor's SEARCH/REPLACE function.

Provided these two rules are observed scrupulously, almost any machine program can be modified in this way. Since I normally use Condensed and Italics, instead of super- or sub-script, I have altered the normal Quill prompts.

Once you have completed the alterations, write the program back to a file under another name and run it. Check through the various commands and options to ensure that the displayed prompts are what was intended and to check that the program is still entirely functional.

Once satisfied that all is as it should be, delete Quill from the disc/cartridge — a back-up copy, of course — and rename your modified file.

I have received several letters, all with the same fatal problem — "Error 103 — Wrong file type." Files which return this error are permanently corrupted and no media recovery system will be of use because the problem is not caused by media corruption

but, instead, by Archive corrupting the data.

Because of the way Archive manipulates files, unless your media is write-protected, Archive files are corrupted as soon as you OPEN the file and make any kind of alteration to it. To this I can say only that, as a continuing operation procedure to help prevent this problem, you can do two things. Never OPEN a file unless you intend to alter it. Instead, always input the LOOK command. Also it is a good idea to get into the habit always of issuing the CLOSE command, even if you are only LOOKING. Also, make back-ups frequently. If, as can sometimes happen, your QL hangs, you will have to re-enter the data only for the most recent time period. The only hope for recovering most if not all of the records in such a file is to use the PDQL *Recover*.

Two readers are involved in writing massive documents and have asked if it is possible to overcome the Quill 254-page limit. The answer is no. The problem is that Quill will not accept a starting page number greater than 254 but will continue to increment from that figure if a document starts there.

To test this, I set the start page at 254 and typed-in a long paragraph and did a block copy until I ran out of memory — 640K at the time. Quill continued to increment page numbering to something around 640.

If you wish to use Quill for writing novels or a lengthy thesis the most practical solution is either to page number in sections or chapters and organise the document so that the last section starts at page 255.

A Belgian reader wishes to have five or 10 more translates than Quill normally provides to use with a wide range of accented characters. Provided bold, underline, sub- and super-script are not required, each of the four pairs of switches can be used as character translate functions, giving access to eight more.

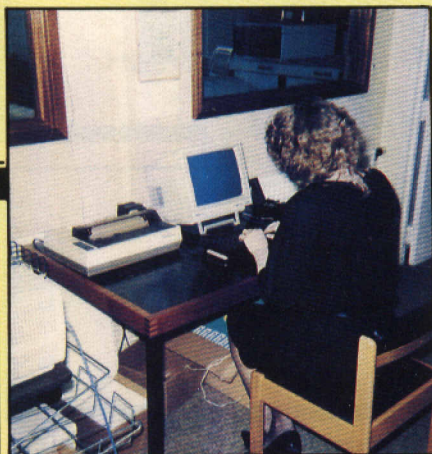


# NETWORKING

I wish that I had learned how easy networking is. I run a Magistrates' Court. I needed to introduce word processing and other desk-top office facilities. When colleagues were choosing the conventional route of obtaining the IBM PC or clones, I decided to go the inexpensive way by obtaining several Sinclair QLs and scattering them round the office.

At first I had one to which initially I interfaced a Brother daisywheel typewriter via an IF50 serial interface. I soon added 256K memory expansion and a double 3.5in. disc drive and obtained a Seikosha SP1000A NLQ dot matrix printer.

A second 512K machine was soon added. Printing initially was achieved by



spooling prints to a Microdrive and transferring that to the machine with the printer. At that point I discovered a useful trick. Since a QL to which no printer is attached will lock up if, in Quill, you opt inadvertently to print a file to "printer", it is important to ensure that all print files are spooled.

That can be guaranteed by using Install\_\_bas to install a "parallel or non-standard" printer and declaring the device to be, say, "Mdv2\_\_prt\_\_lis". If such a file does not exist an attempt to print to "printer" will not hang up the machine but will return the error "Cannot open file", if the file exists it will be over-written.

Despite those fairly primitive arrangements, the demand among my staff for machines on which to work grew. I added an Epson laser printer to improve the quality of output while saving the cost of obtaining printed stationery. I also bought two more 512K workstations, another of which had a disc interface and a double disc drive — one 3.5in. and one 5.25in.

## THE QL

John Davies has put QL networking into practice in earnest. He describes his experience.

On each machine I had *Taskmaster* and on the working *Taskmaster* medium I renamed the "Boot" file as "tBoot" and added a boot file of my own — see below. This was designed to make life as easy as possible for my office staff. What it does is:

- Installs Ramdisc software.
- Formats a Ramdisc.
- Copies to that Ramdisc "printer\_\_dat" and any standard letterheads, spreadsheets or other files which would be

needed on that station.

- Executes *Taskmaster* Boot file (tBoot).

*Taskmaster* loaded on each 512K machine one version of Quill, two of Abacus — sharing Code — and one of Archive. On non-disc drive machines this was set up so that Quill and Archive were loaded from mdv2\_\_ and Abacus — together with the *Taskmaster* utility programs — from mdv1\_\_ to make life as easy as possible; the user had only to feed named media alternately into mdv1\_\_ and mdv2\_\_ as each drive stopped running.

Config\_\_bas was used to set up the Psion programs to look for System Information i.e., printer\_\_dat and data — i.e., def\_\_tmp — on Ram1\_\_ there be relieving the pressure on the Microdrives.

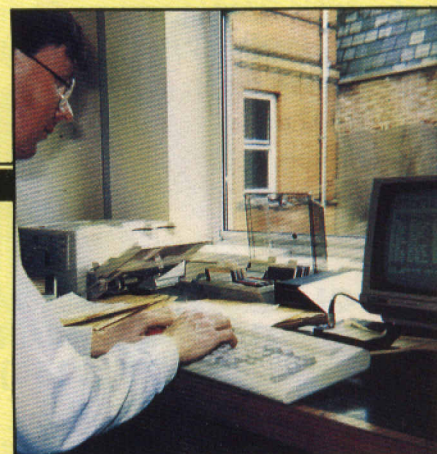
At that stage my two-storey office had two machines on each floor. On one floor the machines are linked by Quadraprint and share the laser printer. Downstairs there was one machine with the Seikosha and one which still spooled its prints to Microdrive.

Using Quadraprint required a degree of co-operation, since if both stations tried to print together they were not queued but the printer got both signals and printed gibberish until it became too confused and locked up. It soon became normal to hear cries of "Printer free?" "O.K." echoing round the office.

The spooling of prints to Microdrive

also had one disadvantage; if one used the same medium more than once to transfer data between the same two machines without re-formatting, one could find that the receiving machine did not recognise the existence of files added since the medium was last inserted in that particular drive. Nothing disastrous would happen but it could be irritating to have to take the directory of a different cartridge before being able to access the files one wanted.

I began to look where I would go when the supply of QLs eventually dried up. Would I be able to add Thors to my set-up? If I wanted to include the driveless workstation I would clearly have to look at networking. I made enquiries and was advised that all I needed was *Toolkit II* by



Tony Tebbby supplied by QJump. I enquired and discovered that I would need the ROM version since RAM is not fast enough to support networking. I ordered three, since I was, at the same time, upgrading the machine with 256K additional RAM by giving it a 640K Trump Card which includes *Toolkit II*.

When they arrived I set two machines down side by side. I connected them by one of the short network cables provided with the QL. I plugged *Toolkit II* into each of the ROM ports and, after a brief pause to read the instructions I went to one machine and typed-in:

NET 1 <ENTER> — the SuperBasic command which gives the machine its network identity and  
FSERVE <ENTER> — a *Toolkit II* program which enables the machine to act as a fileserver.

I put a Microdrive in mdv1\_\_ and went to the other machine. At the other machine I



typed:

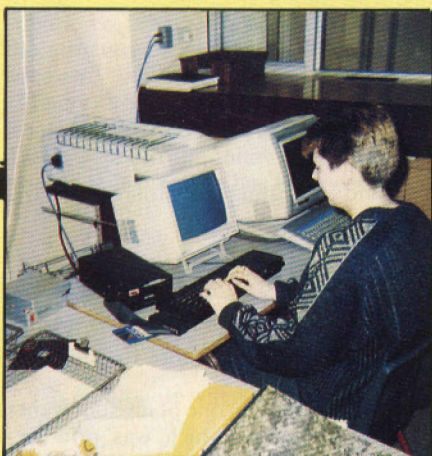
```
NET 2 <ENTER>
NFS__USE
MDV,N1__MDV1__,N1__MDV2__
<ENTER> — a Toolkit II function
which re-directs devices through the
network.
```

I then typed in:

```
DIR MDV1__ <ENTER>
```

To my delight — and I confess surprise — mdv1\_\_ on the first machine whirred and its directory appeared on the monitor attached to the machine on which I was working. I then loaded Quill successfully through the network, although at first I thought it had failed because there was a fairly long pause after the Microdrive had stopped whirring before the opening screen appeared. Those who are accustomed to load these programs from floppy disc will find that it takes much longer to do so through the network. The time is very comparable to loading from Microdrive.

The first thing to do was to install some



wiring. I was able to do it very cheaply; 100 metres of loudspeaker wire from a hi-fi shop cost £15 — 20 pence per metre less 25 percent and eight 3.5mm. jack plugs cost only a few pence each. Loudspeaker wire is recommended since it is low-impedance, twin-core wire like bell wire, recommended in the QL manual but it has in addition a black line marking one core; this is important, since polarity must be maintained between the machines.

A few minutes with a pair of wire strippers and a soldering iron at each end and the job was done; the most time-consuming job was finding ways into the hollow skirting and over the suspended ceiling. Inside the jack plugs one connection was long and one short. I wired the core with the black (L)ine to the (L)ong connection.

Once I had the four machines connected I began to learn a few things about how to set up the network. First I had a disappointment. I had hoped to have on each remote machine nothing more than a

small Boot program along the following lines:

```
10 NET n
20 NFS__USE f1p,n1__f1p1,n1__f1p2__,
n2__f1p1__,n2__f1p2__ etc
30 FSERVICE
40 LRUN f1p1__tBoot
```

I tried this and found that the copy protection of Taskmaster prevented it being executed through the network. Attempts to do so resulted in the "This copy was not produced from a Master copy" error. It was therefore clear that all the Taskmaster files would continue to have to be on a device physically present at each machine. Fortunately I discovered that a working copy of Taskmaster can be made through the network, so it is not necessary to go to a machine which has both a f1p and an mdv to create on the latter a working Taskmaster copy.

Then I discovered two things about NFS\_\_USE. First I found, in attempting to achieve consistency in how devices were addressed from any machine, that it is not possible to include a machine's own drives in the parameters. If you were to attempt the following:

```
NFS__USE f1p,f1p1__,f1p2__,n2__
f1p1__,n2__f1p2__
```

attempts to save to f1p3\_\_ or f1p4\_\_ would save successfully on station 2 but attempts

to use f1p1\_\_ or f1p2\_\_ would be unsuccessful; one's own floppy drives would be inaccessible from one's own machine.

Fortunately, however, the device names given do not need to be actual devices so it is possible to use, e.g., Dev (for Device) or Rem (for Remote). So a machine which has its own floppies and Microdrives can load and save from/to Dev1\_\_ to Dev8\_\_,f1p1\_\_ and f1p2\_\_ and mdv1\_\_ and mdv2\_\_.

One thing you cannot do is achieve a situation where you have one set of network devices called, say, Dev1\_\_ to Dev8\_\_ and, at the same time on the same machine, another set called Rem1\_\_ to Rem8\_\_. The second use of NFS\_\_USE would wipe out the first set. It is possible however, to have up to eight network devices and they can include 'directory' devices:

```
Dev1__ can be n1__f1p1__
Dev2__      n1__f1p2__Jad__
Dev3__      pn1__f1p2__abc__
```

Myfile\_\_doc saved to Dev2\_\_ would be saved on Station 1, f1p2\_\_ as jad\_\_myfile\_\_doc and if saved to Dev3\_\_ it would be saved on the same medium but as abc\_\_myfile\_\_doc.

I have taken advantage of this to try to limit the extent that the media get 'silted up' with old unwanted files. On each

station, except the file server, Dev3\_\_ and Dev4\_\_ are set as n1\_\_f1p2\_\_sN\_\_we\_\_ and n1\_\_f1p2\_\_sN\_\_mo\_\_ respectively. In these assignments 'N' is replaced by the Station number. The result is that if myfile\_\_doc is saved on dev3\_\_ from station 2 and a file of the same name is saved on dev4\_\_ from station 5, the directory of station 1, f1p2\_\_ will be found to contain:

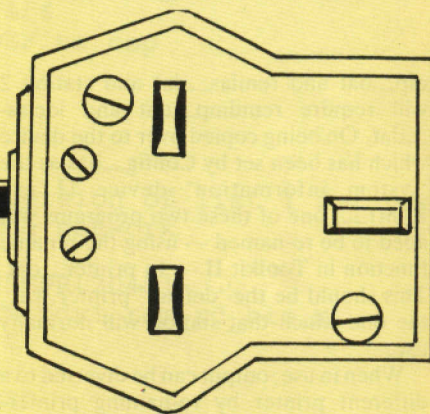
```
S2__we__myfile__doc
s5__mo__myfile__doc
```

The intentions are that:

a. Dev3\_\_ should be used for very temporary material required to be retained for only a (we)ek and Dev4\_\_ for slightly longer-lived material retained for only a (mo)nth. Further, since these files would also be sub-divided by the station from which they were saved, it will be possible to check with the 'owners' before deleting material.

b. Each station should be able to save and over-write files without fear of destroying someone else's work.

Let us suppose that a network of four



QLs has two printers, one a dot matrix printer and one a laser printer, both with serial interfaces and attached physically to stations 1 and 2 respectively. Each printer will require two printer drivers, the files called printer\_\_dat, one with ser1\_\_ as the device and the other (created by using the parallel or non-standard option in install\_\_bas) with nN\_\_ser1\_\_ as the device name where N equals 1 or 2 depending on which printer is attached to which station.

Clearly those four files cannot all be saved on the main file server under the same name and so I give them mnemonic names:

```
remlas__dat — for the laser printer accessed through the net
loc las__dat — for the laser printer accessed directly
rem dmp__dat — for the dot matrix printer through the net
loc dmp__dat — for the dot matrix printer directly accessed.
```

Each station will require two of these files available to it. Stations 3 and 4 will each require remlas\_\_dat and remdmp\_\_dat, station 1 will require locd-



LEGAL AID  
Name: L T Lamb  
19/05/1989

INCOME  
Net In (Excl)  
ALLOWANCES  
Spouse  
Child  
Other  
Hou  
Ref  
Loc  
Tr  
Ot  
CAPITAL  
I  
(  
n  
ALREADY PAID  
LIMITS  
CONTRIBUTIONS

KINGSTON UPON THAMES MAGISTRATES' COURT  
J.A. DAVIS, R.D.  
Barrister at Law  
Clerk to the Justices  
Telephone: 01 546 5603  
Your Ref:  
My Ref:  
Date:  
The Chief Clerk,  
The Crown Court,  
Canbury Park Road,  
Kingston upon Thames,  
Surrey.  
Dear Sir,  
Regina -v- L. T. Lamb  
I enclose for your consideration the trial of Mr. L.T. Lamb Shepton on 14th April, 1989.  
Yours faithfully,  
J.A. Davis  
Clerk to the Justices

19, High Street,  
Kingston upon Thames,  
Surrey.

KINGSTON UPON THAMES MAGISTRATES' COURT  
RECORD OF DECISION WITH REGARD TO BAIL  
Date: 19th May, 1989.  
Accused: L T Lamb  
Of: The Farm Shepton  
The Accused is on bail.  
Name Address and Occupation of Witness  
P.C. Plodd The Police Station Shepton England  
Police Constable 123  
Mr. MAYOR The Mayor's Parlour Shepton England  
Mayor of Toytown

Part II - Statements  
Order Full Page No. 1 - 2  
Full 4 - 6

Income	Weekly Amount	£0.00	Total	£0.00
Capital				

file being accidentally left open on the fileserver: this could occur if a remote station had to be re-set while it was using a file through the network. This situation can be cured by removing and re-executing Server. This can be done by going into SuperBasic on the fileserving machine — <ALT> 9 from Taskmaster

mp\_dat and remlas\_dat and station 2 will require remdmp\_dat and loclas\_dat. On being copied over to the device which has been set by Config\_bas as the 'system information' device (I use Ram1\_) one of these two programs will need to be re-named — using the rename function in Toolkit II — to printer\_dat. This should be the 'default' printer, i.e., the one which that station will normally use.

When in use, output can be diverted to a different printer by re-naming printer\_dat as something else and re-naming the 'spare' printer driver as printer\_dat, if you are using Taskmaster or SuperBasic. This will be made particularly easy if the names you use differ only in the last letter, e.g., 'printer\_dat', 'printer\_dat1' and 'printer\_datd'. If you are using one of the Psion quartet by itself the same can be achieved by copying printer\_dat to a different name, using Files Backup, and the copying the 'spare' printer driver to printer\_dat over-writing the former file.

Earlier when dealing with the use of Quadraprint I explained how it was necessary to ensure that two stations did not try to use the printer at the same time.

## No problem

With the network this is no problem. If an attempt is made to send output to a printer which is in use the error "cannot open file" will be returned. Even this can be avoided by setting up a printer buffer on each machine to which a printer is directly attached. This is done with the command:

```
prt_use ser,ser,ser
```

which will set up buffers on both serial ports; this can be included in a program. Any print files sent to the printer when it is in use will be queued in memory until the printer is free. A revised version of my boot program:

- establishes network devices
- copies various 'standard' files, including printer drivers, through the network
- Sets up printer buffers and
- executes Taskmaster

On a machine which has Trump Card or any other expansion which contains Toolkit II rather than having Toolkit II in the ROM slot, this will need to start with an extra line:

```
1 TK2_EXT
```

to 'switch on' Toolkit II.

Toolkit II includes a program, Server, which must be executed on any machine which is to act as a fileserver. In practice I tend to execute it at every station so that, if desired, files can be transferred between any pair of devices anywhere in the system in either direction. This program is executed by the command "FSERVE", which once again may be included in a program.

Care must be taken as to when precisely it is executed since, if there are any programs which make use of the RESPR() SuperBasic Function, Server cannot be running when the function is called. As a result, I leave it as late as possible and include it in the Taskmaster boot program, re-named as tBoot, immediately before the line which EXECs Taskmaster.

Network problems may be caused by a

— and executing the following direct commands:

```
RJOB Server  
FSERVE
```

This will generally restore the situation.

On one occasion one of my staff went to a remote station which was printing a lengthy document from Archive during which my secretary had 'queued' three letters in the buffer. Thinking that the remote machine had 'frozen' for some reason, she re-set it. The printer stopped; nothing came out. I did RJOB Server, FSERVE and out came my secretary's three letters; the system is remarkably resilient.

## Extravagant

There is only one potential area of difficulty and this concerns Quill. This program had a working file 'def\_tmp' held on the device configured by Config\_bas for Data. Thought must be given as to what device to use so that different copies of Basic at different stations in the network are not all trying to use the same file. One way this could be done would be to have a variety of differently-configured Quills, one for each station. At about 53K of disc space a time this is a comparatively extravagant way of doing it. Two more sensible methods would be:



a. Configure Quill so that the data files are held on a device which is accessed directly from each machine e.g., Ram1\_\_\_, and not through the network from a shared medium or

b. Configure Quill so that def\_\_tmp is held on, say, Dev4\_\_\_ and use nfs\_\_use to ensure that at each station Dev4\_\_\_ is a different directory device, e.g., n1\_\_flp1\_\_sN\_\_\_ where N is the station number.

If it is desired to multi-task two versions of Quill on the same machine there is no option but to have two differently-configured versions of the program — Quill and Quill2 — holding their data files on different devices. This is a limitation which applies only to Quill and not to Abacus, Archive or Easel. I have not tried this but it may be possible to get away with two similarly-configured versions of Quill, possibly code sharing in Taskmaster if one is used for background printing of saved files so that it never has to access def-tmp.

There is a limit to how long a directory device name you can assign by nfs\_\_use; the limit seems to be reached with about two levels of 'sub-directory', e.g., n1\_\_win1\_\_s3\_\_mo\_\_\_ is about as far as you can go. If care is taken in thinking the file structure in advance this should be adequate.

There is some inconsistency between what is acceptable to the Psion suite and to Taskmaster. In the following examples I assume that nfs\_\_use has been employed to ensure that:

```
Dev1___ is n1__flp1___
Dev2___ is n1__flp2___
Dev3___ is n1__flp1__s3__we___
Dev4___ is n1__flp1__s3__mo___
```

The Psion suite will not accept as a valid filename anything which does not begin with a pattern like "AAAnn\_\_\_" where A represents a letter and n a number. In consequence, an attempt to load, say, 'n1\_\_flp1\_\_s3\_\_mo\_\_\_myfile\_\_doc' will be rejected with the error report "Not a valid Quill file." Fortunately, however, the suite of programs will accept 'Dev4\_\_\_myfile\_\_doc' and load 'myfile\_\_doc' correctly from the directory device 'n1\_\_flp1\_\_s3\_\_mo\_\_\_'.

Taskmaster works a little differently. As the "From" device it will accept either the form "n1\_\_flp1\_\_\_" or the form "Dev1\_\_\_". For the device name, however, it clearly reads no further than this and whether one selects Dev1\_\_\_, Dev3\_\_\_ or Dev4\_\_\_, one gets the same complete list of files listed on n1\_\_flp1\_\_\_ listed as, for example:

```
QUILL
remlas_dat
s3__we___myfile__doc
s4__spread__aba
s5__mo___mylet__doc
```

This list can be narrowed only to the directory device dev3\_\_\_ by selecting files

This list can be narrowed only to the directory device, dev3\_\_\_ by selecting files which contain "s3\_\_we\_\_\_" in their titles. Since Taskmaster does not permit two string searches simultaneously this is a regrettable limitation.

It can also be a little confusing since, if one is using Taskmaster, say, to remove some unwanted files, the use of a directory device — dev3\_\_\_ or dev4\_\_\_ — will not work. I suppose what happens is that if you try to remove the third file from the above list from Dev3\_\_\_, what you will try to remove is s3\_\_we\_\_\_s3\_\_we\_\_\_ myfile\_\_doc from n1\_\_flp1\_\_\_ and no such file exists. Unfortunately, however, to add to the confusion, the screen display will react as if the file had been deleted but looking again at the directory will indicate that it has not.

Another little foible is that if one is using the "Choose a file" option in Taskmaster to select a file for, say, Quill and assuming one has chosen the device "Dev3\_\_\_" and chosen from that s3\_\_we\_\_\_myfile\_\_doc, the filename which will appear on the command line on return to Quill is "n1\_\_flp1\_\_s3\_\_we\_\_\_myfile\_\_doc" which is, of course, "Not a valid Quill file." One can edit it to "Dev3\_\_\_myfile\_\_doc" and acquire the file that way but it is a little irritating.

Finally, if one uses the software re-set facility in Taskmaster on a machine which has Trump Card installed, pressing F1 or F2 when the Sinclair copyright screen appears achieves nothing. The machine appears not to recognise the presence of any devices. One has to re-set the machine further using the re-set button. These little details apart, the system is remarkably easy to use and flexible. It is a pity this versatile machine has been sold so short.

## List 1

Program to ease boot up for non computer literate office staff

```
100 m = RESPR(2048)
110 LBYTES mdv1_ram_rd_bin,m
115 CALL m
130 FORMAT ram1_200
140 PRINT "Copying to Ramdisk
    ..."
150 cop "flethead doc",1
155 cop "lethead doc",1
157 cop "llethead doc",1
160 cop "findef aba",1
170 cop "laser dat",1
180 cop "printer dat",1
190 cop "default_aba",2
195 cop "bcisbal_aba",2
200 cop "PAGE_DOC",1
210 LRUN mdv1_tboot
220 DEFINE PROCEDURE cop
    (f$,f)
230 PRINT,f$
240 COPY "mdv"&f&"_"&f$ TO
    "ram1"&f$
250 END DEFINE cop
```

## List 2

Modified version of list 1 for use through the network

```
100 m = RESPR(2048)
110 LBYTES mdv1_ram_rd_bin,m
115 CALL m
120 NET 4
125 NFS USE
    dev,n1__flp1___,n1__flp2___,n2__flp1___,n
    2__flp2___,n2__flp2__s4__we___,n2__flp2__s
    4__mo___,n2__ram1___
130 PRT USE ser,ser,ser
135 FORMAT ram1_200
140 PRINT "Copying to Ramdisk
    ..."
150 cop "flethead doc"
155 cop "lethead doc"
157 cop "llethead doc"
160 cop "findef aba"
170 cop "remlas dat"
175 RENAME ram1_remlas_dat TO
    ram1_printer_dat1
180 cop "nlqprint dat"
185 RENAME ram1_nlqprint_dat
    TO ram1_printer_dat
190 cop "default_aba"
195 cop "bcisbal_aba"
200 cop "PAGE_DOC"
210 LRUN mdv1_tboot
220 DEFINE PROCEDURE cop (f$)
230 PRINT,f$
240 COPY "dev1_"&f$ TO
    "ram1"&f$
250 END DEFINE cop
```



**Information**  
**Program:** Coursemaster  
**Supplier:** Intraset Ltd, 6  
 Gilderdale Close, Gorse  
 Covert, Birchwood, War-  
 rington, Cheshire WA3  
 6TH.  
**Price:** £14.95 inc. p&p. plus  
 £1 for cartridge.

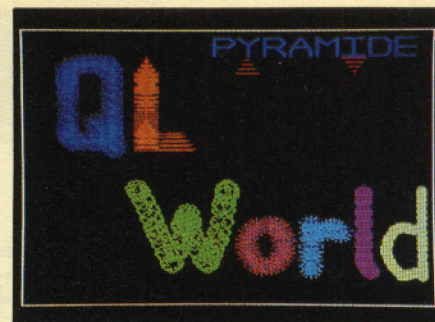
*Coursemaster* is a computer racing program which aims to improve your chances of showing a profit on your bets. It is on Microdrive cartridge only but works on expanded QLs. The package includes a small-format 20-page manual and betting guide.

Horses for courses, the saying goes, but will Coursemaster make you a horsemaster? Andrew Shepard finds that picking a winner is not so easy.

Sinclair/OL World July 1989



# OF MICE AND ART



**D**igital Precision *Eye-Q* set a very high standard for QL art programs. The few art programs which have been released since its appearance in late 1986 have had a tough act to follow. It is almost inevitable that other graphics programs are compared to what has become virtually the industry standard for the QL.

Eidersoft produced a graphics program, *ArtICE*, to accompany its ROM-based *ICE* front-end in the early days of the QL. Soon after the appearance of its first art program, Eidersoft released a new version of *ICE* which included the first and long-awaited mouse. That was soon followed by a mouse version of *ArtICE* called, appropriately enough, *MICEart*. In spite of its long-term availability, *ArtICE* and its mouse-orientated variant, *MICEart*, has never graced the pages of *QL World*.

Much more recently Schön, best-known for its range of quality keyboard replacements, has released its second software product, *Painter*, this time aimed directly at the graphics enthusiast.

In many ways, these art programs share a number of similarities; with their respective versions — they can be used with or without a mouse — but the Eidersoft programs must be ordered in the standard or mouse version, while *Painter* can be used either way.

The Eidersoft *ICE* ROM is a GEM-type environment front-end. The *ICE* screen, displayed every time the QL is re-set unless you hold down the ALT key while the QL is doing its memory check, is provided with icons and buttons which, when the cursor is moved over them and either the SPACE bar is clicked or the mouse button is pressed, the appropriate

**R**on Massey puts on his beret and investigates a sophisticated new art program, *Painter*, and a good old one, *MICEart*.

command is entered, as if it had been typed-in.

Since *ICE* uses its own pointer system it was almost inevitable that a mouse would be produced for it eventually. In a very few months Eidersoft introduced a mouse version of *ICE* called, appropriately enough, *MICE*.

The Eidersoft art program is available as either of two distinct types, both of which require the *ICE* ROM and its integral pointer environment, for use with keyboard only and for use with the mouse version of *ICE* only. To save some confusion, the Eidersoft programs will be referred to as *MICEart* because, except for one of them being mouse-compatible, both programs are otherwise identical.

The Eidersoft mouse is distinctive in that it has three buttons. In effect, the left button is equivalent to pressing the space bar once; the centre button, clicking SPACE twice in rapid succession. The right button is used for special program applications; *MICEart* uses the right button to select solid ink colours sequentially, beginning at black and progressing to white.

On starting *MICEart*, you are presented with the drawing screen and a large disc-

shaped brush. If you wish to start drawing, clicking the left button toggles the brush on. Access to the comprehensive main menu is made by clicking the centre button.

The lack of help pages in any form is mute testimony to the ease of use of the program. Ample prompts are provided throughout operation of the program, advising you of alternative choices or the next action required.

Pressing the centre button, you are presented with the main menu. From that you can navigate to other program options such as selection of ink and paper colours, file access, mode control, magnify, spray, text, clearing the screen and exiting the program.

Colour selection for ink or paper, the only laborious feature of the program, is made by clicking on INK or PAPER, then one set of arrows to set the first colour, a second set of arrows to set the colour and a third set of arrows to set the stipple pattern — if a stipple is used. Numerical values in the range of 0 to 255 are also indicated for information.

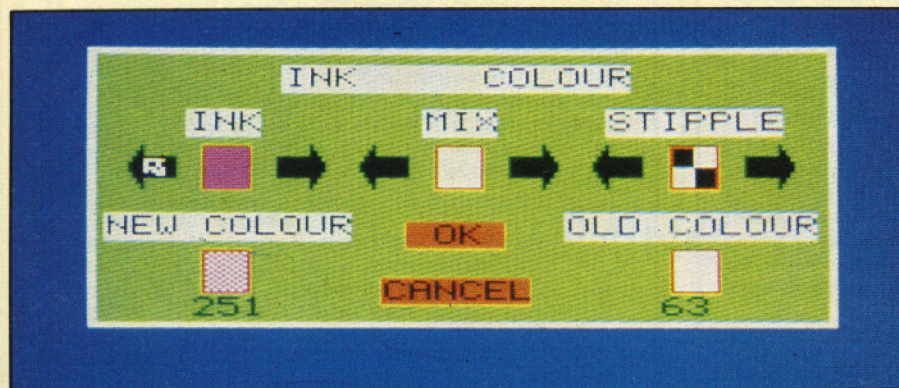
## Brushes

Drawing is done with the cursor pen/brush which is toggled on and off with the left button. A range of brush sizes and types is available, from a single pixel, three discs, four diagonal lines or four blocks, one of which is user-definable, with which to draw.

The *MICEart* magnification facility is particularly good. When active, the area round the cursor is magnified in a mobile window occupying about one-quarter of the screen. As the cursor is moved towards the window, the window changes screen position.

Another particularly well-designed option, spray painting, is especially flexible. You are provided with five different-sized nozzles and five spray densities. The full colour range is available to each of the respective modes.

Text, in the standard QL font and range of sizes, is typed-in and can be positioned accurately by moving the cursor, which appears as the string you have typed-in, to its position and fixing it by clicking on the left button. The majority of drawing operations can be made on a trial basis. After most of the drawing operations, a dialogue box offers you a yes or no option



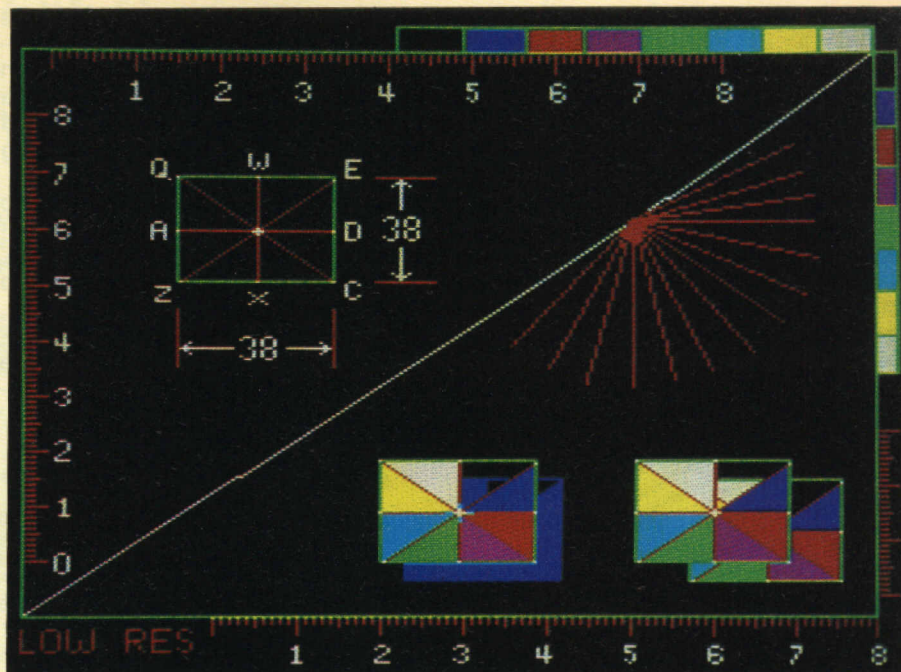


— effectively an UNDO option. Selecting the latter, the drawing reverts to its pre-operation state. Program output can be to printer and an uncompressed, standard 32K file. The MICEart printer driver is a conventional Epson-compatible type.

Being driven by the QJump Pointer environment with which it is supplied, Painter can be used with or without a mouse, multi-tasks and is fully-compatible with QJump QRAM. If you want to take advantage of its mouse capabilities you will also need the QJump QIMI internal mouse upgrade or one of the newer versions of the Sandy Super Q Board, with its built-in mouse interface available.

Painter starts with its menu screen, which consists of five groups of commands and six command buttons. The right-hand side of the screen is a window in which most of the program sub-menus appear. There is only a single help page which details the use of the keyboard for program control. Switching between the menu and the drawing screen is done by pressing ENTER or the right mouse button. Many of the commands are self-explanatory but some exceptional features have been included in the extensive repertoire.

All automatic geometry is rubber-banded. Under the SHAPES heading you have the option either of a rectangle or a square. The latter option will maintain equal vertical and horizontal alteration in



size; the former's dimensions are set independently. The Painter arc feature relies on the flawed QL command and can be a little unpredictable. Line drawing may be done by single lines, radial lines — where lines are projected from a common centre — or as polygons.

On first examination the POINT option seems a little frivolous but the feature

serves a special function. There are 16 types of patterns available, ranging from a single pixel to a variety of pre-set shapes which can be used for a pen-drawing cursor. There is a distinct difference between the Painter PEN and BRUSH. You can select from 24 supplied brush patterns to draw. You can also design your own brushes or use a 16 × 16 pixel

## Eidersoft MICEart

Drawing Method	Continuous; pixel graphics.
Definition Modes	Movement into pixel position. 4 and 8; loss of picture at mode change.
Multiple screens	No.
Transfer Image Elements	No.
Method of colour selection	See Note 1.
Command Access	Mouse.
Menu	Yes; main and sub-menus.
Help pages on screen	None.
Border reference	No.
Grid	No.
Cursor co-ord indicator	No.
Prompt Window	With some options.
Image Pan/Scroll	No.
Image magnification	Yes.
Auto mirror image	No.
Pen direction indicator	No.
Pen (Continuous drawing)	Yes.
Width control	No.
Brush	Single pixel; 3 discs; 4 diagonal; 4 blocks, one of which can be re-defined.
Airbrush	5 nozzles and 5 densities.
Auto fill — on select	Also separate manual fill.
Expand	No.
Shrink	No.
Erase	Area depending on brush size and shape.
Special erase	No.
Undo	With most drawing operations.
Pen On/Off	Toggles with left mouse button.
XOR	No.
Re-colour	Yes.
Circle	Yes.
Ellipse	No.
Arc	No.
Square	Horiz and vert dimensions independent.
Rectangle	As above.
Triangle	No.

Others	Block.
Line length	Yes.
Line width	No.
Line broken	No.
Radial lines	No.
Polygon	No.
Point	Cursor-shaped.
Element movement	No.
Element copy	Can also be saved and/or used as a brush.
Auto Shadowing	No.
Text	Yes.
Modes	Over current drawing and Off.
Colour	Full range.
Sizes	0,0 TO 3,1 (dependent on resolution mode).
Character styles	Standard QL font.
Variable spacing (auto)	No.
Underline types	None.
Positioning	Text string cursor.
<b>File Control</b>	
Default drive	No.
Retain other specified drive	No.
Directory	From load, save and copy to a file.
Load a screen	Yes.
Save a screen	Yes.
Compression option	No.
Delete a file	No.
Format media	Yes.
Load/Save/Edit fonts	No.
Brush patterns	None.
Printer Dump	Yes.
Area printed	Whole screen.
Average no. files/cartridge	10.
Principal application	Two-dimensional illustrations.

### Special Features:

1. Ink is selected by clicking the right button sequentially switches ink through black to white, colour numbers 0 to 7 or from the main menu, where the sub-menu offers buttons which are clicked to change each of the three components making up a stipple. Paper colour from the sub-menu.
2. A non-mouse version, ArtICE, is entirely keyboard-orientated and uses the SPACE bar instead of mouse buttons. Requires ICE ROM.
3. Text is positioned as if the string were a cursor.



cursor to select an area from your drawing to use as a brush. Full file handling facilities allow you to save and load individual brush files into the space occupied by a current brush.

Two types of eraser are provided. The first is the conventional type which is consistent with the selected cursor. For larger and/or regular areas the special eraser is a rubber-banded rectangle.

Colour selection may be done in either of two ways. Solid colours may be selected from the main menu palette buttons. By double-clicking on the stipple button you are offered the sub-menu palette from which you can select a colour directly either from the full range palette display or, very much like the MICEart selection system, select each colour component with its appropriate stipple pattern.

One Painter feature which puts it in a class of its own is its ability to have up to nine screens available — memory permitting — and being able to transfer sections from a selected page to the current drawing.

Clicking on VIEW, the display switches to a map of all the screens in memory and a pixel representation of the drawings of each screen. You are offered an option to create a new page. A word of warning — the create option is not error-checked and if you specify a number of pages exceeding your memory capacity you are likely to

hang the QL. File options use the QRAM convention, with a similar range of commands, and a specifically-efficient screen compression routine which squeezes a standard 32K screen into between 4K and 6K of file space. Saving and loading compressed screens is very fast.

A number of choices allow you to produce good-quality dumps of your pictures. You can select the whole screen, half or a selected part of the screen to dump. You can also specify single-, double- or treble-density printing. Pictures can be positioned either over an entire sheet of paper or can be printed sideways.

## Transfers

For different reasons I like both of these programs. I am not particularly fond of mice and, as far as I am concerned, art programs are the only real justification for using them. Having said that, though, screen navigation and option selection in all types of mouse-controlled programs is faster, so it is really a matter of preference.

Which of the two programs you choose will depend largely on the way you like to work. If you prefer having an icon/button-orientated front-end, ARTice or its counterpart, MICEart, is the logical choice.

Neither of the two programs can be very readily multi-tasked.

Painter is designed to be multi-tasked with all the QJump utilities and uses the same pointer environment. If you are running QRAM you will not need to use the separate pointer environment supplied with the program.

Like many art programs available for the ST and Amiga, MICEart allows you to select only one option each time you access the main menu. Its spray facility must rank as highly as any of those available for other machines. The Painter spray facility is simulated in that a drawing cursor producing the "spray" is a fixed but re-definable pattern.

Even if you are not artistically inclined, the Painter text editing features make the program worth having for this one application alone. In addition to including its own font designer you can display the current font in normal or bold lettering and in one of four types of italics.

Characters, occurring from the upper left corner of the cursor position, may be printed as Outlines or Shadow, as well as standard lettering. Choosing the Shadow option you can specify top, left, bottom and right shadowing in any combination.

Overall, my vote goes to Painter. It is one of the most versatile and powerful art programs yet released for the QL. Its only disadvantage is that a few of its operations are a little slow.

## Schön Painter

Drawing Method	Continuous; pixel graphics. Movement into pixel position.
Definition Modes	4 and 8;
Multiple screens	Up to nine, memory permitting.
Transfer Image Elements	From selected screen to current screen.
Method of colour selection	See Note 1.
Command Access	Mouse and/or keyboard.
Menu	Yes; main and sub-menus.
Help pages on screen	1.
Border reference	No.
Grid	No.
Cursor co-ord indicator	No.
Prompt Window	With some options.
Image Pan/Scroll	Yes; whole screen or selected area.
Image magnification	Yes.
Auto mirror image	Yes; rotate in 90° increments.
Pen (continuous drawing)	Yes.
Width control	No.
Brush	24 patterns supplied. See note 3.
Airbrush	Simulated with pixel pattern block.
Auto fill — on select	Also separate manual fill.
Expand	Yes.
Shrink	Yes.
Erase	Area depending on brush size and shape.
Special erase	User definable block.
Undo	With ESC.
Pen On/Off	Toggles with left button/SPACE.
XOR	To current ink colour.
Re-colour	Full-screen or selected area.
Circle	Yes.
Ellipse	Yes.
Arc	Yes.
Square	Horiz and vert dimensions linked.
Rectangle	Horiz and vert dimensions independent.
Triangle	No.
Others	Block.
Line length	Yes.
Line width	No.
Line broken	16 types.
Radial lines	Yes.

Polygon	Yes.
Point	16 footprint patterns.
Element movement	No.
Element copy	Can also be saved and/or used as a brush.
Auto Shadowing	No.
Text	Yes.
Modes	Over current drawing, XOR and Off
Colour	Full range.
Sizes	0,0 TO 3,1 (dependent on resolution mode).
Character styles	Normal, 4 Italics, Bold. See note 2.
Variable spacing	Between characters and proportional.
Underline types	16.
Positioning	From current text cursor position.
File Control	GRAM-type display.
Default drive	f1p1.
Retain other specified drive	Yes.
Directory	Yes.
Load a screen	Yes.
Save a screen	Yes.
Compression option	Yes.
Delete a file	Yes.
Format media	Yes.
Load/Save/Edit fonts	Yes.
Brush patterns	24 supplied. With design, load and save options.
Printer Dump	Yes.
Area printed	Whole, half or part screen; sideways or whole page.
Density	1 to 3.
Average no. files/cartridge	10 uncompressed.
Principal application	Two-dimensional illustrations.

### Special Features:

1. Colour can be selected directly from the main menu by clicking on the palette display. Clicking on the stipple button selects the current stipple.  
Alternatively, double-clicking on the stipple button opens a full-screen sub-menu. It comprises of a large window, containing blocks of all of the colours available to the mode, a pair of solid colour palettes, four buttons representing the four possible stipple patterns and a window displaying the selected colour.
2. Brushes may be composed of one of the 24 supplied patterns, selected from a 16 × 16 pixel area of the current drawing, may be a solid colour or may be loaded into from a pattern file.



**A**lmost every computer programmer has an ambition to write a home finance program which will keep track of cheques, balance budgets and plan future finances. Many such programs are started, some are finished, but I suspect only the smallest minority are every used. The disadvantage of using a computer program to control personal accounts is that it is too powerful for the job. Booting the QL, loading a program and saving a data file just to record the issue of a cheque is not worth the effort.

To make a conventional financial program work for you it must be updated regularly with information. The incessant demand for cheque numbers, amounts drawn, income received, interest added and deducted and standing order renewal dates soon overwhelms the average user's enthusiasm and the program is abandoned in favour of a pencil and the back of an envelope.

Personal budgeting is a perfect example of what a spreadsheet is good at doing and so Abacus was chosen as the tool to implement the *Sinclair QL World Budget Planner*, an application most readers should find useful. Before plunging into the technical aspects of programming a spreadsheet it is worth analysing what happens when planning a budget. Income is balanced with outgoings so that, with luck, the latter does not exceed the former in the long term. Problems occur when either income or expenses, or both, are irregular. Eventually a spate of bills will exceed the available income.

### Staircase

Budgeting means calculating how much to put aside and when. If every bill had to be settled in December a simple budget would involve putting aside one-twelfth of the predicted total debt each month. The balance of the budget

amounts are rounded to the nearest pound and many of the figures inevitably will be estimates. Although the example spreadsheet shows only a few rows, a proper working spreadsheet might have dozens of entries covering gas bills, car servicing, standing orders, insurance premiums, savings schemes and so on to the limit of the QL memory.

The second area comprises mainly a 12 x 12 matrix and it is here that the major calculations take place. Each column represents a staircase of gradually-increasing amounts set aside for each month's expenditure throughout the year.

To create the budget spreadsheet the instructions accompanying this article should be followed closely. It is recommended strongly that an exact copy of the example spreadsheet is created, tested against the output reprinted here and before modifying it to suit your circumstances. It has been assumed that readers know how Abacus works. If this is

**Mike Lloyd employs Psion Abacus to provide a spreadsheet which will predict your cashflow and can be updated quickly**

# BUDGET PLANNER

The most basic home finance programs reveal little information the user does not already know and their claims of pinpoint accuracy usually founder on the calculation of bank charges, interest calculations and rate demands. Most people are not concerned with maintaining records of expenditure or balancing the books to the nearest penny and they do not enjoy slaving at a computer program just to confirm that the latest bank statement is correct.

### Invaluable

People are generally more interested in making sure they can meet the next electricity bill or save enough for a holiday. For this exercise a computer is an invaluable aid and yet few financial software packages tackle the problem of budgeting.

A budget management program could undoubtedly be written in SuperBasic but there is a much simpler solution available to every QL owner, the Psion Abacus spreadsheet. Spreadsheets are very effective at dealing with repeated calculations which form simple patterns based on columns and rows. Spreadsheets are straightforward to program and easy to keep updated. Most important, they are particularly good at processing what-if? enquiries when figures are changed temporarily to see the effect on other parts of the spreadsheet.

account would increase each month so that on a graph it would look like a staircase rising from January to November. In December, of course, the account is cleared to meet the debts and the graph falls to zero.

Reality, of course, is different. Bills arrive monthly, quarterly, annually or irregularly; some bills are for regular amounts and others vary with the seasons. Calculating a savings staircase for each month would be a splendid basis for a sensible budget, except that the effort is probably not worthwhile even with the help of a calculator. The long-winded sums would need to be re-calculated every time a bill was increased or a new commitment was added or a debt needed to be cleared earlier than expected.

With a spreadsheet, however, all the calculations are contained in a few formulae based on amounts placed in the spreadsheet cells. The amounts can be changed as often and as radically as required but the fundamental relationships between them remain the same and the entire spreadsheet can be re-calculated in seconds.

In the Sinclair QL World Budget Planner the spreadsheet is divided into three distinct parts. Beginning from the left of the spreadsheet, the first part contains details of all known debts with their amounts entered in columns according to which month they become due. Budgeting is never an exact science and so

your first spreadsheet, keep the User Guide close to hand.

The three diagrams accompanying this article contain all of the important Abacus commands needed to make the spreadsheet application work. Each command is linked to the spreadsheet cell to which it applies and shows either a formula or an Abacus command. There is no room to include many minor instructions covering such aspects as text justification and column widths. Similarly, text not related to a formula has not been included in the commands. Text can be entered by moving the cursor to a cell and typing a double quotation mark followed by the required letters. The text string is not closed with a quotation mark.

### Integers

As an example for QL users new to Abacus, it will be convenient to change the Abacus numeric input to integers before entering any other commands. This is done by pressing F3 to obtain the command mode and then pressing "U" for the units command, "D" to signify the Default option, "I" to obtain integer input and the Enter key to select a minus sign for negative values. Had this been included in one of the diagrams it would have been written:

F3 Units, Default, Integer, Minus sign

The first area in the spreadsheet has a



**F3 DESIGN A**  
(TURN AUTO-CALC OFF)

B4 980

**F3** ECHO, CELL B4 OVER RANGE C4:M4

B6 SUM(B7:B20)

**F3** ECHO, CELL B6 OVER RANGE C6:M6

B8 320

**F3** ECHO, CELL B8 OVER RANGE C8:M8

OTHER AMOUNTS ENTERED EITHER DIRECTLY OR USING ECHO COMMAND AS ABOVE.

ALL TEXT IN THIS AREA IS ENTERED DIRECTLY

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Sinclair QL World Budget Planner												
2													
3	For 1989	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
4	NET INCOME	980	980	980	980	980	980	980	980	980	980	980	980
5	EXPENSES												
6	TOTALS:	786	716	906	876	1716	796	1276	716	946	736	670	1000
7													
8	Mortgage	320	320	320	320	320	320	320	320	320	320	320	320
9	Rates	46	46	46	46	46	46	46	46	46	46	46	46
10	Water Rates	50					50						
11													
12	General	250	250	250	250	250	250	250	250	250	250	250	250
13	Credit Card	100	100	100	100	100	100	100	100	100	100	100	100
14	Electricity			100			80			60			80
15													
16	Holidays				900		500						
17	Birthdays, etc	20		30	10		10		10	20		250	
18													
19	Car			150	100				160				
20	TV Licence		60										

CELL M20 GRID USE A1:AG20 MEMORY 15K  
CONTENTS EMPTY

THE COLUMNS FOR EACH MONTH CAN BE EXTENDED TO AS MANY ROWS AS ARE NECESSARY WITHIN THE LIMITS OF THE QL'S RAM SPACE. IF MORE ROWS ARE ADDED, YOU MUST MAKE SURE THAT THE TOTALS FORMULAE IN ROW 6 ARE AMENDED TO INCLUDE THE NEW ROWS.

details column with each item entered directly as text followed by 12 columns headed by the name of each month in the year. Your spreadsheet does not have to start with January but it is convenient.

The most important rows are those detailing income and the total expenditure. They must appear in the same rows as shown in the example if the formulae elsewhere in the spreadsheet are to be correct. The income figures indicate that the user is paid the same amount per calendar month. Adjustments would need to be made for people paid daily or weekly amounts or with irregular income.

It is important to note that the income for a month usually relates to what was earned in the previous month. Someone paid on a daily rate would receive 28 days' pay for February at the same rate per diem as for March, a 31-day month.

It would be conventional to place the total monthly expenditure row at the bottom of the columns but the position of

the row would then change when rows are added or taken away. By placing them near the top of the columns the total figures never move their location and the number of rows used can be altered with no alterations to the rest of the spreadsheet.

Monthly expenses which do not vary need be entered only in the January column and then echoed to the other columns for the year, as shown by the entry for mortgage repayments in the diagram. Irregular amounts are entered directly into the appropriate cell. The example spreadsheet also shows how estimates for electricity bills vary to take account of seasonal variations in fuel used.

The second part of the spreadsheet is the matrix of staircases showing that each month's total expenditure has been divided into 12 equal parts and one part is being saved each month. The entire matrix is governed by a single formula

copied into each of the 144 cells. A feature of the matrix which is always present is a diagonal line of zeros crossing the matrix from top left to bottom right. For May, the zero appears in the fifth row of the fifth column, counting from the top left corner of the matrix.

Immediately to the right of the matrix each row is totalled. If they were the required minimum end-of-month balances the budget scheme would probably be unworkable but there is an extra calculation in the third part of the spreadsheet which reduces those figures.

The final part of the spreadsheet is a synopsis of the budget for the year. The months of the year are now arranged down the side of the spreadsheet, with columns for income and expenditure repeating information from elsewhere in the spreadsheet. The most important column is the minimum end-of-month balance which is derived from the matrix totals.

**F3** GRID, WIDTH, 4, FROM 0 TO 2

N6 MONTH(ROW()-5)

**F3** ECHO, CELL N6 OVER RANGE N7:N17

**F3** GRID, WIDTH, 9, FROM N TO N

O6 INT((INDEX(COL()-13,6)/12)+4\*(ROW()-COL()+9+12\*(ROW()-COL()+9,8)))

**F3** ECHO, CELL O6 OVER RANGE O6:217

	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
1	Calculation of Minimum End-of-Month Balances													
2														
3	January	0	649	750	657	1144	462	636	295	312	183	110	83	5281
4	February	65	0	825	730	1287	528	742	354	390	244	165	166	5496
5	March	130	59	0	883	1430	594	848	413	468	305	220	249	5519
6	April	195	118	75	0	1573	660	954	472	546	366	275	332	5566
7	May	260	177	150	73	0	726	1060	531	624	427	330	415	4773
8	June	325	236	225	146	143	0	1166	590	702	498	385	498	4904
9	July	390	295	300	219	286	66	0	649	780	549	440	581	4555
10	August	455	354	375	292	429	132	106	0	858	618	495	664	4770
11	September	520	413	450	365	572	198	212	59	0	671	550	747	4757
12	October	585	472	525	438	715	264	318	118	78	0	605	830	4948
13	November	650	531	600	511	858	330	424	177	156	61	0	913	5211
14	December	715	590	675	584	1001	396	530	236	234	122	55	0	5138
15														
16														
17														
18														
19														
20														

Minimum = 4555 AA19 MIN(AA6:AA17)

CELL N1 GRID USE A1:AG20 MEMORY 15K  
CONTENTS "Calculation of Minimum End-of-Month Balances"



AB6 MONTH(ROW()-5)	AB	AC	AD	AE	AF	AG	AE5 AA17-\$AA19+50
F3 ECHO, CELL AB6 OVER RANGE AB7:AB17	Synopsis of Monthly Income and Expenditure						AG6 AE5+AC6-AD6-AE6
AC6 INDEX(ROW()-4, 4)				END OF MONTH			F3 ECHO, CELL AG6 OVER RANGE AG7:AG17
F3 ECHO, CELL AC6 OVER RANGE AC7:AC17		Income	Expenses	MIN BALANCE	Excess		AE6 AA6-\$AA19+50
AD6 INDEX(ROW()-4, 6)							F3 ECHO, CELL AE6 OVER RANGE AE7:AE17
F3 ECHO, CELL AD6 OVER RANGE AD7:AD17							AG19 SUM(AG6:AG17)
ALL OTHER TEXT ENTERED DIRECTLY							F3 ECHO, CELL AG19 OVER RANGE AC19:AD19
THE APPEARANCE OF THE SPREADSHEET CAN BE ENHANCED BY ADJUSTING ROW WIDTHS, CENTERING TEXT, ETC.							
	1	2	3	4	5	6	
	6 January	980	786	633	51		
	7 February	980	716	776	49		
	8 March	980	906	1014	51		
	9 April	980	876	1061	57		
	10 May	980	1716	268	57		
	11 June	980	796	399	53		
	12 July	980	1276	50	53		
	13 August	980	716	265	49		
	14 September	980	946	252	47		
	15 October	980	736	443	53		
	16 November	980	670	706	47		
	17 December	980	1000	633	53		
	18 TOTALS	11760	11140	620			
	19						
	20						
	21						
	22						
	23						
	24						
	25						
	26						
	27						
	28						
	29						
	30						
	31						
	32						
	33						
	34						
	35						
	36						
	37						
	38						
	39						
	40						
	41						
	42						
	43						
	44						
	45						
	46						
	47						
	48						
	49						
	50						
	51						
	52						
	53						
	54						
	55						
	56						
	57						
	58						
	59						
	60						
	61						
	62						
	63						
	64						
	65						
	66						
	67						
	68						
	69						
	70						
	71						
	72						
	73						
	74						
	75						
	76						
	77						
	78						
	79						
	80						
	81						
	82						
	83						
	84						
	85						
	86						
	87						
	88						
	89						
	90						
	91						
	92						
	93						
	94						
	95						
	96						
	97						
	98						
	99						
	100						

In this example the matrix totals show that the balance of the account would never fall below £4,555, the lowest figure in the column. If, therefore, all of the figures were reduced by £4,555 the account would never be in the red, although for one month it might be empty. As an empty account is not generally a good idea, a safety factor of £50 has been built into the calculations in column AE.

If this final adjustment to the figures seems too good to be true, let me assure you that it is correct and that following the budget will not lead you unexpectedly into trouble. The final column in the spreadsheet shows how much income remains

once the commitments have been met. An advantage of this budgeting scheme is that, provided your income is regular, you will be left each month with a similar amount for incidentals and luxuries.

When the spreadsheet has been built, saved and tested, what next? First, the example amounts need to be replaced with figures applicable to your circumstances. The total income and total expenditure figures at the bottom of columns AC and AD should then be checked to make sure that income exceeds expenditure over the year. You may then wish to return to the first part of the spreadsheet and change some of the figures to see what

effect they have on the synopsis columns.

When the budget has been finalised, take a printout for ready reference when the bank statements arrive. The program needs to be referred to again only if your budget changes or if you want to see the impact of, say, buying a more expensive holiday or repaying a loan early.

I have been using this budgeting system for four years and it has proved its worth. It has identified precisely how much of my bank balance can safely be spent on luxuries and how much must be kept for future commitments. Best of all, the application is free, easy to use and quickly adaptable.

# TF Services

## London STD codes

London 01- dialling code is changing to 071- on 081- in May 1990. If you have a computer based address list, we can provide the data you need to change your lists, and even a program to change text files (eg Psion EXPERT files).

Disks contain 5 text lists in database report format: (quote deleted), etc & a program for text file export file conversion. Also provided:

- IBM PC : Files for major databases/spreadsheets
- QL : Psion Archive .dbr file
- ATARI ST : The 5 text lists & program only

3.5/5.25 disk or microdrive - £10

## COMPUTER CLEANER

Much improved mains filters, 40-90 db cut & 3 spike filters, 390 joule cut (3 & 4 way)

Certified X/Y 240ac capacitors

- 2-way (5a adaptor)..... £14
- 3-way (5a adaptor)..... £18
- 4-way (33a trailing socket)..... £24

## Qualsoft Terminal

Viewdata/VT52/ASCII QL TERMINAL EMULATOR

'Een deluxe communicatieprogramma van Qualsoft'

Multitasking program for electronic mail, PRESTEL etc. Phone director es for ALL types ALL modems for the QL, autodial (where poss) and logon, 50-way FILE TRANSFER to ATARI ST, IBM PC, Psion Organiser (via coas link), VIDEOTEX, real time clock/timer, buffered logs to file, printer, transmit files, editable command line, wordwrap, Norwegian options, ECL translates, editor etc.

SOLVES ALL PACKAGED MODEM SOFTWARE PROBLEMS (unbuffered modems usable with Miracle Modem)

Software (3.5" or ndv) & manual £30

## QUICK QL REPAIRS

LARGEST stock of components & working circuit boards to ensure quick and reliable repair. (Usually same day)

QLs tested with Thorn EMI test rig and ROM software

CMOS hardware extra if required

**£25 (6m gntee)**

## FILE TRANSFER

Programs to transfer text and programs error free (XMODEM) between computers

Available for IBM PC and compatibles, ATARI ST and Sinclair QL. Connects to Psion organiser via Psion coas link

Qualsoft program (per 1/0)..... £7.50

Serial lead (max 2 computers)..... £10

Complete package for 2 computers: £25

All prices include VAT and postage in the UK

Callers welcome (by appointment only please)

12 Bouverie Place, London W2 1RB (tel: 01-724 9053)

Fax: 01-706 2329 Telex: 265451 (ref: 723869045) Prestel: 017259053

# REBEL ELECTRONICS LTD

## AT LAST PERFORMANCE PRODUCTS FOR YOUR QL

### RB 100 — WINCHESTER DISC CONTROLLER (Price £195)

- High performance Western Digital Chipset
- Standard ST506/ST412 drive interface
- 8K byte sector buffer
- Plugs into QL expansion port or backplane
- Controls up to two physical drives

### RB2000 — CONTROLLER + 20M DRIVE + PSU (Price £399)

- Ready to go system
- Elegant styling

### RB50E/RB500E EXPANSION BACKPLANES (£83 or £192 EACH)

ALL PRICES INCLUDE VAT AND P&P

To order: Please fill in coupon and post to

## REBEL ELECTRONICS LTD

12 YORK PLACE, LEEDS LS1 2DS. Or telephone **0757 86630/0904 708073**

PLEASE HAVE YOUR QL CARD READY

Please send me:

RB100 ☐ RB2000 ☐ RB50E ☐ RB500E ☐

I enclose cheque — Value £

or debit Visa Card Expiry date:

Signature: Name:

Address:



# THE NORTHERN SINCLAIR SHOW

**D**avid Batty and Sector Software are to be congratulated for their bravery in organising the first Sinclair show in the north and also on their choice of venue. Stokes Hall at Leyland, Lancashire is a pleasant setting and reasonably-priced food was available. The atmosphere was definitely "microfair" but as a customer I felt more relaxed and human all day.

The show was attended by more than 30 exhibitors, most selling QL-specific or related items. The event was held on two levels with two main rooms full of stalls, the remainder being housed in the foyer. Principal QL suppliers exhibiting their wares were Digital Precision, Sector Software, PDQL, Miracle Systems, TF Service, Strong Computers, Qualsoft and Super User Bureau.

## Throng

Sector Software had the largest and best-sited stall, which was very busy throughout the day. Hidden from sight by a throng of eager customers were the adjacent stalls of Digital Precision and PDQL. SUB was selling copies of the second issue of its club magazine. There were apologies for being about a year behind but I was assured that further issues will emerge. They were also digitising faces and printing them out for £1.50 and had a steady stream of children queueing for the service.

In the foyer TF Services and Qualsoft were side by side and must have won the award for the highest number of working computers on one stall, with QLs, PCs, Psion Organisers and an ST all running next to each other. Miracle Systems attended but had a shortage of stock. It blamed that on underestimating demand for its pro-

**I** Sector Software has founded a QL micro show of a different kind — a show in and for the north. Now users who were daunted by the long trek to the London Microfairs have an alternative. It seems set to be a pronounced success, reports Desmond Barry.

ducts in the previews two months earlier and was talking mainly about its hard disc drive which was on display.

Rebel Electronics, a company new to the QL scene, was displaying a non-working ex-

Home Computers selling daisywheel typewriters which doubled as computer printers for £89. It also had a good stock of new QLs for less than £100 each.

The show was very busy in

should have been at the exhibition but was not ready. Leaflets were available on the system from the Sector Software stand. While talking to the Sector staff I asked for a copy of *Ferret* and was told the disc box had been left open overnight and it had escaped. Then it was admitted that there had not been time to get it ready for the show.

Freddy Vaccha of Digital Precision gave an impressive talk about his new MS-DOS emulator and kept me captivated for more than half an hour.

Harvester Information Systems had new software for the Z-88, including a typing tutor and spelling checker. It was so new it was still being copied during the show.

## Radio

Radio Lancashire had its outside broadcast van in attendance and transmitted live from inside the building, while *Lancashire Evening Post* girls distributed complimentary newspapers specially printed with the Northern Sinclair Show logo and the names of the five companies which sponsored the paper giveaway.

Living in the north of England I was pleased to see that we at last had a show of our own rather than having to travel to London. The majority of people, who appeared all to be local, shared that view. Few of those present had made the journey from the south. One man told me that he and his friends had crossed on the ferry from Northern Ireland.

Afterwards I telephoned Sector Software — 0772 454328 — and was told that as the event was such a success more shows are scheduled for June 24, September and one in December for Christmas.



The throng at Leyland Hall checks new wares.

ample of its hard disc interface which is said to use hard discs of up to 80MB capacity and operates at faster speeds. The Miracle drive will work with the Trump Card but the Rebel drive needs a smaller QL to work.

## Bargains

Impact Entertainments demonstrated a *Trivial Pursuit*-type game and a horse racing predictor. On the non-QL-specific stalls there were also some good bargains, with U.K.

the morning and I found it difficult to get near to many of the stalls but by about 2pm the crowds started thinning and the exhibitors had more time to talk.

A licensed bar was open throughout the exhibition, serving as a meeting place for QL enthusiasts. Food was available in the cafeteria throughout the day, well organised with good cutlery, serviettes and tablecloths, and good value food.

Sir Clive Sinclair's new Cambridge Satellite TV system

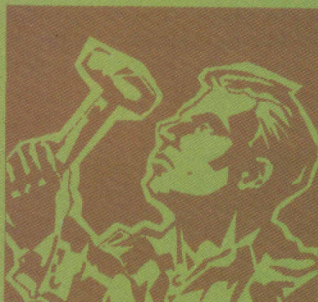


# DIY TOOLKIT

This month  
Simon Goodwin  
adds a useful  
new device  
driver to the QL  
repertoire.

**T**his month's listing is the most sophisticated DIY Toolkit ever — a new device driver, complete in just 414 bytes of code. The program is derived from a listing sent by the two Italian *QL World* readers Fulvio Morsella and Luca Pivat. The MEM device works with all Qdos programs and languages and allows very flexible communication between tasks or program overlays. It works like other devices but can do many things not possible with standard facilities like pipes and shared files.

The MEM device also extends the *Toolkit 2* network server so that you can do almost anything over the net-



work; you can even issue PEEKs, POKEs and calls to routines in the memory of other machines.

Anything written to the MEM device is stored in memory. Anything read from it is fetched from memory. There are no limitations on the data which can be read or written; each character corresponds to one byte and you can use all the usual character input and output commands — PRINT, IN-KEY\$, LBYTES, SBYTES, INPUT\$, GET, BGET, PUT, BPUT and so on.

MEM works with any command which sends characters to a channel; you can send the result of LIST, DIR and similar commands to MEM. Bytes are written and read in sequence towards higher addresses, with no check for the end of mem-

\* QL WORLD DIY TOOLKIT JULY 1989 - MEM device driver,  
\* version 0.9, copyright Luca Pivato & Simon N Goodwin.

```
*
mem_ptr equ 24      Current MEM pointer
buff_id  equ 28      Word ID of buffer
chan_link equ 30     Channel list pointer
buff_addr equ 34     Buffer start address
buff_flag equ 38     Word persistence flag
```

```
*
start lea.l serio_ptr,a0 Set up SERIO linkage
      lea.l io_ready,a2
      move.l a2,(a0)+
      lea.l fetch_byte,a2
      move.l a2,(a0)+
      lea.l send_byte,a2
      move.l a2,(a0)+
      lea.l io_pointer,a0 Set up device linkage
      lea.l io_code,a2
      move.l a2,(a0)+
      lea.l open_code,a2
      move.l a2,(a0)+
      lea.l close_code,a2
      move.l a2,(a0)+
      lea.l linkage,a0 Link the MEM device
      moveq #32,d0 MT.LI00 key
      trap #1
      rts Return D0 to caller
```

```
*
linkage dc.l 0 Link to next device
io_pointer dc.l 0 Pointer to I/O code
          dc.l 0 Pointer to OPEN code
          dc.l 0 Pointer to CLOSE code
```

```
*
buff_ptr dc.l 0 Start of buffer list
chan_ptr dc.l 0 Start of channel list
```

\* Code to handle TRAP #3 calls: IO.PEND, FBYTE, FLINE,  
\* FSTR6, SBYTE, SSTR6, EXTOP, POSAB, POSRE, LOAD, SAVE

```
*
io_code cmp.b #9,d0 Call to FS.EXTOP?
      beq vector,a2
      move.l mem_ptr(a0),a5 A5 -> MEM address
      cmp.b #66,d0 Call to FS.POSAB?
      beq.s set_abspos
      cmp.b #67,d0 Call to FS.POSRE?
      beq.s set_relpos
      move.w 234,a2 Use IO.SERIO vector
      jsr (a2) Do other serial I/O
      serio_ptr dc.l 0 Pointer for IO.PEND
      dc.l 0 Pointer for IO.FETCH
      dc.l 0 Pointer for IO.SEND
      rts Return from SERIO
```

```
*
set_abspos tst.l d1 Negative parameter?
      bpl.s use_posn
      move.l buff_addr(a0),d1 Get default base
      use_posn move.l d1,a5 Set position
      bra.s set_memptr
      set_relpos add.l d1,a5 Offset position
      move.l a5,d1 Return new value
      bra.s set_memptr
      send_byte move.b d1,(a5)+
      bra.s set_memptr
      fetch_byte move.b (a5)+,d1
      set_memptr move.l a5,mem_ptr(a0) Reset MEM pointer
      io_ready moveq #0,d0
      rts
```

\* Code to handle TRAP #2 calls: OPEN and CLOSE

```
*
open_code subq.l #6,a7 Make space on stack
      move.l a7,a3 A3 -> parameters
      move.w 290,a2 Use IO.NAME vector
      jsr (a2)
      bra.s exit_open Name faulty
      bra.s exit_open Parameters faulty
      bra.s open_ok Name parsed OK
      dc.w 3 Length of name
      dc.b 'MEM'
      dc.w 3 Max. 3 parameters
      dc.w -1,-1 Buffer number
      dc.w ' ',-1 Buffer size
      dc.w 2,'PT' Permanent/Temporary

*
open_ok move.l a3,a5 A5 -> parameters
      moveq #40,d1 40 bytes needed
      move.w 192,a2 MM.ALCHP vector
      jsr (a2) Allocate memory
      beq.s do_buffers Go on if RAM permits
      exit_open addq.l #6,a7 Deallocate stack space
      tst.l d0 Error code is in D0
      rts
```

```
*
do_buffers move.w (a5),d7 Get buffer ID
      move.w d7,buff_id(a0) Keep buffer ID
      bmi.s exit_open No buffer, exit
      set_flag move.w 4(a5),buff_flag(a0) Keep P/T flag
      move.l a0,a4 Save channel base
      lea.l buff_ptr,a3 Search buffer list
      bsr.s scanner Does the buffer exist?
      beq.s new_one No it doesn't
      lea.l 6(a3),a0 A0 -> Start of buffer
      moveq #0,d0 OPENed without errors
      bra.s set_addr Tell the channel
```

```
*
new_one move.w 2(a5),d1 Was a size specified?
      bmi.s no_size If not, complain!
      addq.w #8,d1 Allow for a header
      ext.l d1 D1 = total buffer size
      moveq #0,d2 Permanent allocation
      moveq #24,d0 MT.ALCHP key
      trap #1 Try to allocate memory
      tst.l d0 Did that work?
      bne.s no_room If not, complain
      move.l a0,d2 Keep buffer base
      move.w d7,(a0)+ Record buffer ID
      lea.l buff_ptr,a3 Extend the buffer list
      move.l (a3),(a0)+
      move.l d2,(a3) Start with new buffer
      sub.w #24,d1 Ignore header bytes
      move.w d1,(a0)+ Store buffer length
      set_addr move.l a0,buff_addr(a4) Record buffer start
      move.l a0,mem_ptr(a4) Initialise pointer
      lea.l chan_ptr,a3 Extend channel list
      move.l (a3),chan_link(a4)
      lea.l buff_id(a4),a2
      move.l a2,(a3)
      move.l a4,a0 A0 -> Channel block
      bra.s exit_open
```

```
*
no_room move.l a4,a0 Retrieve channel
      bsr.s lose_chan
      moveq #-3,d0 No room for the buffer
      bra.s exit_open Report OUT OF MEMORY
```



```

no_size bsr.s lose_chan
        moveq #15,d0      Size needed but absent
        bra.s exit_open   Report BAD PARAMETER
*
scanner move.l a3,a2      Find the ID in D7.W
        move.l (a2),d0    Try the next link
        beq.s not_found   No such luck, quit
        move.l d0,a3      A3 -> Buffer ID.W
        cmp.w (a3)+,d7    Is it what we want?
        bne.s scanner     If no, try the next
found_it tst.l d0         Yes, return A2 & A3
not_found rts            Return D0=0 if absent
*
close_code lea.l buff_id(a0),a4 Get list position
           lea.l chan_ptr,a3   Purge channel list
           bsr.s purge_list
           move.w (a4),d7      Is a buffer in use?
           bmi.s lose_chan     No, just zap channel
           subq.w #1,buff_flag(a0) Is buffer permanent?
           beq.s lose_chan     Yes, just zap channel
           lea.l chan_ptr,a3   Is the buffer busy?
           bsr.s scanner      Search channel list
           bne.s lose_chan     If busy keep buffer
           move.l buff_addr(a0),a4 Find buffer
           subq.l #8,a4        Include header
           lea.l buff_ptr,a3   Scan buffer list
           bsr.s purge_list    Un-link buffer
           exg a4,a0           Swap pointers over
           moveq #25,d0        MT.RECHP key
           trap #1             Deallocate buffer
           move.l a4,a0        Restore channel base
lose_chan move.w 194,a2       MM.RECHP vector
vector_a2 jmp (a2)           Deallocate channel
*
purge_list move.l a3,a2      Remove link to (A4)
           move.l (a2),d0
           beq.s bizarre     No more - abort!
           move.l d0,a3      A3 -> Buffer ID.W
           addq.l #2,a3      Skip buffer ID
           cmp.l a4,d0       Have we found it?
           bne.s purge_list  No, look further
           move.l (a3),(a2)   De-list the entry
bizarre   rts
*
end

```

ory, so you can use MEM to read ROMs or write to control ports if you wish.

INPUT works, so long as you know there is an end-of-line marker - CHR\$(10) - a little later in memory. Beware; in the course of testing this code I have found another bug in Qdos and the Thor Argos.

Bizarre things happen if the interpreter tries to INPUT a line of more than 32,766 bytes. If you read such a line into an undimensioned string you may have a string more than 32K in length; 32K is the official limit - longer strings have an apparently negative length and tend to crash the QL.

I will explain this in more detail in my next report on ROM bugs, as I keep finding new ones. For the time being, be warned not to use INPUT unless you know there is an end-of-line marker among the next 32K characters. Use Turbo Toolkit INPUT\$ if you need to read a certain number of

bytes, regardless of their value.

There are several ways to use the MEM device. The simplest technique is to open it with no parameters, like this:

OPEN #3,mem

This sets up channel #3 so you can read and write directly into memory. You set the address being read with common Toolkit commands like PUT or SET\_\_POSITION, used normally to wind back and forth through a file.

PUT #3/131072 or SET\_\_POSITION #3,131072 positions the channel pointer at the start of the standard QL display memory, explored in the last issue. A subsequent PRINT #3,FILL\$("?!"); 16384); over-writes the top half of the screen with a stripe pattern.

When you OPEN a simple MEM device the memory pointer starts at address 0 but you can move it easily with PUT or

the Turbo SET\_\_POSITION. If you do not have those comms you can set the pointer with devious calls to SCROLL and PAN, as I explained in my last ROM bugs article, in the February *QL World*. Functions like POSITION and FPOS tell you the current position of the pointer.

So far, the MEM device is a neat example of a new device and a useful way to access memory directly from languages which lack PEEK and POKE but it is rather a low-level facility unlikely to appeal to people who do not PEEK and POKE already.

For a start you can open MEM channels via the network file server built into Thors and SuperToolkit ROMs. This gives you direct access to the guts of any machine serving the network with MEM loaded. The simple MEM device works perfectly over the network. I have tested it with two QLs and between a QL and Thor XVI.

### No hacking

You can use all the normal reading and writing commands; PUT and FPOS let you keep track of the current address on the other machine.

You can have as many MEM channels open as you like. It is easy to avoid malicious hacking as you cannot access the memory of a machine unless it is running the file server with MEM loaded.

There is great potential for co-operative processing. You can store code on another machine via MEM and then call it with the EXTOP TRAP. When the code has finished you can read the results from MEM. Thus you can circumvent limitations of the normal file server and change modes or format media remotely.

If this romping through memory seems messy, try tacking a few parameters on to the end of the device name. MEM allows up to three parameters - an integer buffer number and buffer size, and a letter 'p' or 't' to show whether the buffer is 'permanent' or 'temporary'. This command opens a 2,000-byte permanent buffer, number 7:

OPEN #3,mem7\_2000p

Once you have done this you can read or write the buffer



from other tasks or the same one like this:

OPEN #4,mem7

When you open a MEM buffer the file pointer starts to set to the beginning of the buffer - not zero, as for a simple MEM device. There is no check to prevent you writing past the end, as all MEM channels can access any address, but it is easy to add checks from Basic or in the device code. The word immediately before the start is the buffer size, in bytes:

PRINT "Buffer size = ";PEEK\_\_W(FPOS(#4)-2)

You can wind back to the start with SET\_\_POSITION #ch%,-1 or PUT. If you are stuck with the standard QL ROM commands, try:

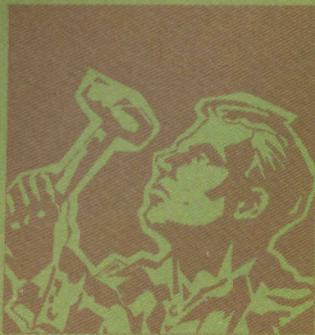
SCROLL #3,-1,42

Buffers can be shared by many channels and, if appropriate, they can all read and write them at once. The clever piece is the way any program can find the address of the buffer from the name, via the MEM device driver.

In the past it has been difficult for programmers to pass information like device defaults between programs, especially if the programs load one after another, to make best use of RAM. Many products use 'spare' system memory, from 163,876 upwards, to pass information but clashes occur when several products try to use the same area for different purposes. This is such a common problem that it has become a major source of incompatibility between QL programs, regardless of supplier.

Supercharge used screen memory, which could be upset by multi-tasking windows. Spellbound, Desktop Publisher





and *SuperToolkit* fight it out over undocumented system variables. *Turbo* used a buffer after the documented system variables but this caused a clash on the Thor XVI, ironically because the undocumented memory was used for a pointer to a 'thing list'.

A 'thing list' is a sequence of buffers linked by pointers. Its use avoids clashes between communications areas but each package needs code to search and extend the list. Unfortunately you still need a system variable to point to the start of the list and that is what clobbered the original version of *Turbo* on the Thor XVI. QJump and CST propose different starts for this list and it seems inevitable that both are already used for other purposes.

The MEM device is my solution. It is simple, fast and uses the device list, so it clobbers no system variables. MEM allows 32,768 buffers, numbered from 0 to 32,767, but there is still the possibility that two packages might try to use the same buffer. If MEM buffer numbers clash you can change them easily by patching the device-names in one group of programs, with *Spy* or *The Editor*. It is easy to identify and change device names because they are ASCII strings.

Buffer memory is de-allocated automatically when the last channel using it is closed. If that channel was opened with a 'p' at the end of its name the buffer persists so that it can be opened and read later, perhaps by another task. You get rid of a 'permanent' buffer by waiting until no channel is using it, opening it as temporary – the default – and closing it.

The MEM device should work reliably on any QL or compatible, including the Thor and ST QL Emulator. It was tested on version PT of the

Thor XVI and the JS and MG versions of the QL ROM.

The only problem I have found so far concerns the use of numbered buffers over the network. I tested this aspect with Toolkit 2.12 on my QL and MEM running on a Thor XVI file server.

I ran into difficulty after opening a new buffer in Thor memory from the QL. Then I opened it on the Thor, which found it without problems. For some unknown reason it was not possible to read the correct buffer address with FPOS from the QL, although random ac-

cess to Thor RAM worked satisfactorily. The QL gets an address of zero over the network, even though the Thor finds the buffer and gives the correct address every time.

The same thing happens if I open the buffer on the Thor first, then try to link to it from the QL. I tried setting a position of -1 but the pointer still seemed set to zero. I could not set any position less than zero or above 16 million-odd.

It seems that the top eight bits of the value are lost en route over the network. I am not sure whether this problem

lies in the Thor, the QL or MEM. I shall investigate further and hope to explain all next month.

In any case this quirk does not invalidate the technique. The buffer idea is most useful when communicating between programs on a single machine; you can network the address of a pre-allocated area easily enough by writing the address to a file. Other machines can read the address over the network with OPEN\_IN, once the file has been created and closed by the machine which owns the buffer.

The code for the MEM device is listed in two forms. Listing one is the assembly code program, assembled using HiSoft *DevPac*. You can type this text into your assembler if you want to customise the MEM device or merge it with other codes.

Listing two gives you a quick way to enter the code without using an assembler. It loads the equivalent machine code from DATA statements and saves the code in a file. Once you have loaded that file, as follows, you can OPEN MEM channels from any task which runs on that machine:

base=RESPR(414):LBYTES  
"file name",base:CALL base

The first part of listing two is Marcus Jeffery's standard loader, used in each month's DIY Toolkit project. Only the DATA, from line 590 onwards, changes from month to month, so that is all you need to type if you have typed-in a DIY Toolkit listing previously and remembered to save it.

The MEM device is the most sophisticated DIY Toolkit routine to date and uses several techniques which have never been explained properly in books about Qdos. There is no room to explain the listing or give detailed examples this month.

● In the next issue I will be back with a detailed code commentary, diagrams and short programs which show how MEM can be useful. I am still eager to hear what you would like to see in this column. Please send your suggestions if you would like me to explore a specific area, or implement routines which seem useful but have never appeared in commercial toolkits.

```

100 REMark Sinclair QL World HEX LOADER
110 REMark by Marcus Jeffery & Simon N Goodwin
120 :
150 CLS: RESTORE : READ space: start=RESPR(space)
160 PRINT "Loading Hex..." : HEX_LOAD start
170 INPUT "Save to file..." : f$
180 SBYTES f$,start,byte : STOP
190 :
200 DEFine FuNction DECIMAL(x)
210 RETurn CODE(h$(x))-48-7*(h$(x)>"9")
220 END DEFine DECIMAL
230 :
240 DEFine PROCedure HEX_LOAD(start)
290 byte = 0 : checksum = 0
300 REPeat load_hex_digits
310   READ h$
320   IF h$="" : EXIT load_hex_digits
330   IF LEN(h$) MOD 2
340     PRINT"Odd number of hex digits in: ";h$
350     STOP
360   END IF
370   FOR b = 1 TO LEN(h$) STEP 2
380     hb = DECIMAL(b) : lb = DECIMAL(b+1)
390     IF hb<0 OR hb>15 OR lb<0 OR lb>15
400       PRINT"Illegal hex digit in: ";h$ : STOP
420     END IF
430     POKE start+byte,16*hb+lb
440     checksum = checksum + 16*hb + lb
450     byte = byte + 1
460   END FOR b
470 END REPeat load_hex_digits
480 READ check
490 IF check <> checksum
500   PRINT"Checksum incorrect. Recheck data.":STOP
520 END IF
530 PRINT"Checksum correct, data entered at: ";start
560 END DEFine HEX_LOAD
570 :
580 REMark Space requirements for the machine code
590 DATA 414
600 :
610 REMark Machine code data
620 DATA "41FA006C45FA0092","20CA45FA008620CA"
630 DATA "45FA007C20CA41FA","002245FA003220CA"
640 DATA "45FA007A20CA45FA","012C20CA41FA0008"
650 DATA "70204E414E750000","0000000000000000"
660 DATA "0000000000000000","000000000000B03C"
670 DATA "0009670001362A68","001B803C0042671C"
680 DATA "B03C004367223479","000000EA4E920000"
690 DATA "0000000000000000","00004E75A816A04"
700 DATA "222800222A41600C","DBC1220D60061AC1"
710 DATA "6002121D214D0018","70004E755DBF264F"
720 DATA "3479000001224E92","6026602460140003"
730 DATA "4D454D000003FFFF","FFFF205FFFFF0002"
740 DATA "50542A4B72283479","000000C04E926706"
750 DATA "5CBFA4B04E753E15","3147001C6BF2316D"
760 DATA "00040026284847FA","FF5E6158670841EB"
770 DATA "000670006026322D","00026B42504148C1"
780 DATA "740070184E414A80","662C240830C747FA"
790 DATA "FF3620D326820441","001830C129480022"
800 DATA "2948001847FAFF24","2953001E45EC001C"
810 DATA "26BA204C609A204C","614A70FD60926144"
820 DATA "70F160BC244B2012","67082640BE5B66F4"
830 DATA "4A804E7549E8001C","47FAFE0612E3E14"
840 DATA "6B2253680026671C","47FAFEE061D66614"
850 DATA "28680022518C47FA","FECE6110C14C7019"
860 DATA "4E41204C34790000","00C24ED2244B2012"
870 DATA "670A2640548B08C","66F224934E75","*",32408

```



**C**omputer users struggling with poor screen displays can suffer from eyestrain, headaches and even back problems. There are three ingredients to poor screen displays — dot crawl, glare and reflections. While there is no universal cure for these problems it is rare to find all of them occurring at once. It is important, though, to identify which of the problems is present and then to apply the correct solution. Purchasing the cure for a problem you do not have might increase your troubles as well as waste your money.

Dot crawl occurs only on TV displays and is acceptable provided that the screen is viewed from a distance of at least one metre, that the computer is not used for long periods and that high-definition work is not undertaken. TV displays are therefore unsuitable for graphics design, desk-top publishing and extensive word processing. Dot crawl can be minimised by selecting, where possible, double width character sizes — at least CSIZE 2.0 — using the QL eight-colour low-resolution mode and by avoiding some colour combinations. The only cure for dot crawl is to replace the TV set with a monitor.

## Glare flare

Glare is a by-product of poor screen contrast, which encourages users to increase the brightness setting until the bright parts of the screen display "flare". Glare might be so bad that halos round characters such as "m", "n" and "w" make them indistinguishable. A partial improvement can be obtained by locating the monitor away from any strong light source, such as a window, and by darkening the room, thus allowing the monitor brightness setting to be reduced. Desk lights should be positioned slightly behind the monitor screen and well to one side. If those measures are ineffective, screen contrast can be enhanced greatly by the addition of a polarising filter to the front of the monitor.

Polarised sunglasses have been available for many years. They take advantage of the fact that light waves oscillating vertically are responsible for

# Anti Glare Screen

**Mike Lloyd reports on screens to reduce computer eyestrain.**

much more glare than light waves oscillating horizontally. Polarised lenses have a molecular structure which blocks much of the vertical light waves and let through most of the horizontal light waves. That is why using polarising sunglasses while lying on your side is worse than not wearing them.

Glare could be reduced by wearing non-tinted polarised glasses when computing but a more practical solution is to buy a relatively inexpensive polarising filter mounted on a firm plastic frame which is attached to the front of the monitor by means of Velcro patches. Not only is the contrast between light and dark pixels enhanced but the richness of colouring is improved.

This is particularly noticeable with red ink on a black background, a combination usually unreadable on monochromatic screens. Various sizes of frame are available to suit most makes of monitor but polarising filters have one big disadvantage — their shiny surfaces increase the problem of reflections.

External reflections are the biggest single cause of visual discomfort when using a computer monitor. Except by taking the extreme step of controlling the intensity and position of all light sources in the computing area, little can be done to prevent annoyingly intrusive reflections from degrading the screen display.

The Microvitec Cub, one of the most popular monitors for the QL, is particularly prone to reflections. Some monitors have a ground glass screen which dissipates much of the reflected light but this also reduces the display contrast and so contributes to glare.

A more recent development is the mesh filter, also known as the silk screen filter, which can be made either of a special and very expensive glass or from a finely-stretched piece of non-reflective nylon mounted in a plastic frame. Do-it-yourself fanatics might like to try stretching an old pair of

women's tights over their monitors.

Anti-reflection filters work on the principle that reflected light usually hits the screen at an angle. The mesh only allows through light which is perpendicular to the face of the screen, thus giving light from the monitor an advantage while absorbing light from external sources. Again, these filters are in many sizes to suit different monitor designs and they are often described, confusingly, as being anti-glare filters.

## Cub filter

I have been trying a mesh filter for the Microvitec Cub bought from Immac, which supplies by mail order to business customers. It costs £30 plus VAT and is in a simple protective carton. The frame is coloured black on one side and light grey on the other so that it could complement either the QL or the BBC variants of the Cub. Included in the package are four small Velcro strips which are fixed to the monitor surround by very strong double-sided sticky tape. The filter needs to be removed occasionally to clean the screen.

The Cub has a distinctly curved front and the filter is uncompromisingly straight; therefore the only attachment points were at the top centre and bottom centre. The increased distance between the screen and the filter at the edges tended to make characters look very slightly out of focus. The filter does not remove all reflected light because the nylon threads of which it is made are not completely non-reflective. In strong light, the diamond grid of the nylon weave can clearly be seen but the overall effect is remarkable.

A little care is needed to keep the filter in good condition. Glass and polarising filters can be cleaned in the same way as screens but

nylon mesh filters would be damaged by sprays and cloths. They are cleaned by a little roller of double-sided tape which picks up pieces of dust.

All filter types can be bought with an anti-static device if static electricity is a particular problem but for most users that is an unnecessary sophistication. Smokers can soon destroy a mesh screen by inadvertently touching the nylon with a lighted cigarette. Clumsy users risk putting their fingers through the mesh if they point to things on the screen.

People soon become so accustomed to the improved display that they need to remove the filter occasionally to remind themselves of how bad the reflections used to be. My company is now buying 80 mesh filters because everybody wants one.

Prices for filters depend on the supplier, the size of the monitor, the advertiser's perceived market and the quantity ordered. Polarised filters tend to be less than £20, while mesh filters cost about £25-35. Glass mesh filters are available for around £100. Check with your supplier for details about costs for filters suitable for your monitor and ask for advice about what kind of filter to obtain.

## INFORMATION

### Polarising filters

Advantages: Improves contrast, enhances colours, cheap.

Disadvantages: Shiny surface can cause reflections.

Target Price: £12-£15

### Nylon mesh filters

Advantages: Reduces external reflections, some reduction in glare.

Disadvantages: Not very robust. Slightly awkward to clean.

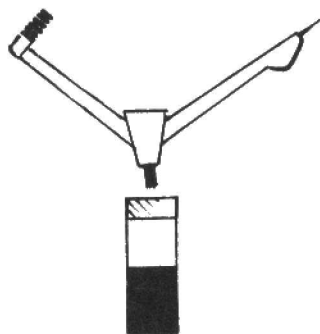
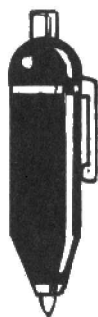
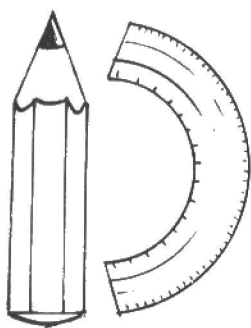
Target Price: £25-£35

### Glass screen filters

Advantages: Removes up to 99 percent of reflections and improves colour contrast. Robust. Easy to clean.

Disadvantages: Expensive  
Target Price: £90-£120.





# ARC

## In part two of his series on Archive programming, John Davis looks at procedures for copying and opening files and creating dbf files.

**A**ny program will need procedures to do the housekeeping — copying, opening and closing of files. It is sensible therefore to have a suite of procedures which will do this. Before moving on to write these it is necessary to say a little about variable names. First it is important to remember that the reserved words — the commands provided as part of Archive — may not be used as variable names, even with the suffix '\$' as a string variable.

Parameters and variables declared as 'local' will be recognised only within the procedure to which they apply. They will not affect or be affected by global variables — those applying to the whole program. As an example, go into edit mode and write the following procedure:

```
proc adder;a
  local b
  let b = 10
  let c = a + 10
endproc
```

Escape from edit mode and type:

```
let a = 100
let b = 200
let c = 300
adder;2
print a
print b
print c
```

The results should be 100, 200 and 12 respectively; the global variables a and b are unaffected by the parameter a or the local variable b.

One word of caution is necessary. Archive looks for parameters and local variables before it looks for global variables but it looks for fieldnames before it looks for either. So if you have the following procedure for helping format print files:

```
proc 1p;t;p$
  1print tab t;p$
endproc
```

and a global variable p\$ which contains the value "Jim" and you enter the follow-

ing command 1p;10, "Fred", the word "Fred" will be sent to the printer in column 10. If you then open a \_\_dbf file which has

a field p\$ the value of which in the current record is "Mary" and once again enter the command:

1p;10, "Jim"

much to your dismay you will discover that the word "Mary" will be sent to the printer. Close the file and try again and once more "Jim" will appear in print. Finally enter the command:

1 Print tab 10;p\$

and the word "Fred" will appear at the printer; the global variable has been unaffected by the parameter with the same name.

This can be very disconcerting; it took me several weeks to discover what was going wrong when my procedure "ops" for opening a number of files always came up with an error after opening a particular file. Since many of one's procedures may well be general-purpose and likely to be used with a number of different \_\_dbf files, I believe it to be useful to have a convention for ensuring that the two never conflict. As a general rule I now reserve variable names of one or two characters +\$ for strings — for parameters and local variables and I use names of three characters or more for fields and global variables. I commend the practice.

It is inevitable that any program will involve copying even if only to make a back-up copy of the amended \_\_dbf at the end. To avoid failing with an error any previous copy of the file must be deleted before the back-up copy is made. The following procedure does both:

```
proc cop;o$n$,d1$,d2$
  kill d2$+n$
  backup d1$+o$ as d2$+n$
endproc
```

o\$ is the file to be copied, n\$ is the new file name, d1\$ is the source drive and d2\$ is the drive to which the copy is to be made. So the command cop;"myfile","myfile","mdv1\_\_\_","mdv2\_\_\_" will copy the file called myfile from mdv1 to mdv2 with the same name and

### Listing 1

A program to create \_\_dbf files for a hypotheticalal database with files "d", "o" and "w"

```
proc init
  let ind1 = 9999999
  let ind2 = 9999999
  let ind3 = 9999999
  append
  close
endproc
proc makd
  kill prg$+"d"+" dbf"
  create prg$+"d" logical "d"
  x$
  ind1
  etc
  n$
  endcreate
  init
endproc
proc makemain
  kill prg$+" dbf"
  create prg$+" logical "d"
  wdr$
  sdr$
  scdr$
  budr$
  files$
  levels$
  screens$
  menus$
  prints$
  endcreate
  append
  alter
endproc
proc makeo
  kill prg$+"o"+" dbf"
  create prg$+"o" logical "o"
  x$
  ind1
  etc
  n$
  endcreate
  init
endproc
proc makew
  kill prg$+"w"+" dbf"
  create prg$+"w" logical "w"
  etc
  endcreate
endproc
proc start
  rem Change name to "make" if
  included in main program during
  development
  input "Programme Name?";prg$
  error makemain
  error makd
  error makeo
  error makew
endproc
```



# CHIVING

"cop","a","b","mdv1\_\_","mdv1\_\_" will copy file a as file b on the same drive.

It is obviously good practice to keep an unopened copy of the file on which you are working, since a system crash with a \_\_dbf file open may well cost you the contents of that file. If you are using Microdrives and have both this working copy and the security copy on the same medium it will be prudent to have yet another copy on another medium.

## Ramdisc

If you have Ramdiscs available it is a good idea to keep your screen files on Ramdisc to improve response times. Provided the files are not too large you can also keep the open working copy of your \_\_dbf files on Ramdisc as well for the same reason. If you have a large memory expansion but are still using Microdrives you may well have more space on your Ramdisc than on any one mdv. If you do this, remember to close and copy promptly as soon as you have done your work as data on Ramdisc is very vulnerable; copying back to a permanent medium every half-hour or so during a work session is not a bad idea. Another risk is that you could end up with a \_\_dbf file on the Ramdisc larger than 110K which you would be unable to copy to mdv without deleting some records.

As supplied, Archive allows you to choose where to keep your system information — the program + printer\_\_dat — Help files and data files. The Superbasic program config\_\_bas is used to change these.

Although it is not immediately obvious in Archive, 'data' files do not just mean \_\_dbf files; the term also includes \_\_scn, \_\_prg and \_\_pro files. You may wish to keep these on different devices and to be able to vary on which device you keep particular types of data depending on whether you have, for example, Ramdisc or on the size of your \_\_dbf file. As a result I identified the need for four global variables to hold the identities of four different devices:

wdr\$ — holds the open working \_\_dbf files  
sdr\$ — holds the security copy of the \_\_dbf  
schr\$ — holds the screen files  
buds\$ — holds any additional backup copy of the \_\_dbf files

Also, if these procedures are to be general-purpose and available for a variety of programs, it is necessary to have a variable called prg\$ to identify the files relating to a given program. Since you may wish your programs to be easily converted to run on PC Four on an IBM to have a variable called join\$ which is an underscore for the Qdos-based machines and a full-stop for IBM PCs.

Concerning the naming of files, I decided that:

a. Programs should have four character names.

b. Working \_\_dbf files which, in a sense are ephemeral only, should be given single-character names.

c. Working screen file should have two-character names, the first character identifying the file to which it relates and the second identifying the particular screen.

d. Working copies of menu screen files should have a name resulting from the concatenation of "men" and a single character.

e. The main copies of all these should have a name concatenated from the program name and the working file name.

So a program called "comm" with four \_\_dbf files called "d","w","o" and "z" for each of which there is only one \_\_scn file and only one menu has the following files:

```
held on sdr$
commd__dbf
commw__dbf
commo__dbf
commz__dbf
held on wdr$
d__dbf
w__dbf
o__dbf
z__dbf
held on sdr$
commda__scn
commwa__scn
commoa__scn
commza__scn
commenu__scn
held on sdr$
da__scn
wa__scn
oa__scn
za__scn
menu__scn
```

By using the block copying convention I found it possible to concatenate all the various \_\_dbf file names together into a

global variable called files\$, all the screen names into a variable called screen\$ and the menu names into a variable called menu\$. For the foregoing program the values of these variables are as follows:

```
Files$      dwoz
Screen$     dawaoaza
Menu$       u
```

I then do all necessary copying using the following procedures:

```
proc opcop
rem makes working copies of __dbf files
local no
let no=1
while no<len(files$)+1
  cop;prg$+files$(no)+join$+
  "dbf",files$(no)+join$+"dbf",s-
  chr$,wdr$
  let no=no+1
endwhile
cstat
endproc
```

```
proc scrcop
rem makes working copies of screens
use "main"
local po
let po=1
while po<len(screens$)+1
  cop;prg$+screens$(po to po+ 1)+
  join$+"scn",screens$
  (po to po+ 1)+join$+"scn",schr$,
  schr$
  let po=po+2
endwhile
let po=1
while po<len(menus$)+1
  cop;prg$+"men"+menus$(po)+
  join$+"scn","men"+menus$
  (po)+join$+"scn",schr$,schr$
  let po=po+1
endwhile
cstat
endproc
```

```
proc clocop
rem copies back __dbf files at end of
session
stat;"Making Security Copies"
use "main"
local n
let n=1
while n<len(files$)+1
  cop;files$(n)+join$+"dbf",prg$+
  files$(n)+join$+"dbf",wdr$
  $,schr$
  let n=n+1
```



```

        endwhile
        cstat
        endproc

proc bucops

    rem makes extra copy of __dbf files on
    different medium
    stat,"Making Backup Security Copies"

    if budr$=sdr$
        return
    endif
    local n
    let n=1
    while n<len(files$)+1
        cop;prg$+files$(n)+join$+"dbf",
        prg$+files$(n)+join$+dbf
        ",sdr$,budr$"
        let n=n+1
    endwhile
    cstat
    endproc

```

In the foregoing procedures the procedure "stat" prints the string given as a parameter centre justified on the bottom line of the screen — assuming a monitor with a screen width of 80 columns in reverse video.

```

proc stat;p$
    local t$
    ink 0: paper 7
    if len (p$)>80
        let t$ = p$(1 to 80)
    else
        let t$ = p$
    endif
    print at 21,40 — int(len(p$)/2);p$
    ink 7: paper 0
    endproc

```

The following procedure clears this 'status line' by printing a line of 80 spaces — white ink on black paper — over the message displayed by "stat":

```

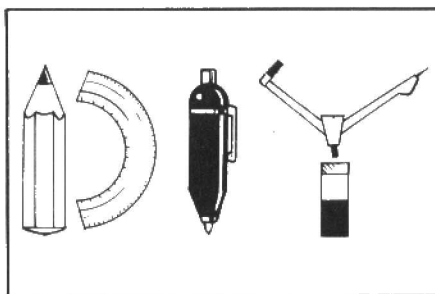
proc cstat
    print at 21,0;rept(" ",80)
    endproc

```

Those filenames above "com-ma\_\_dbf", "a\_\_dbf" and so on are not the only names files have; each open \_\_dbf file must have a logical filename (lfn). If no files are open already with the lfn "main" and a file is opened without any lfn, Archive will allocate it automatically the name "main"; otherwise it is up to the programmer.

When a number of files are opened the last one will be the 'current' file — the file used by a number of commands unless, in some cases, another is specified. The current file can be changed by the command "use lfn". If "a" is the current file and you wish to Search, Find, Locate or Insert — all commands which can be executed only on the current file — on "b":-

use "b"



will achieve what you want. The following commands can be executed on a file other than the current file if an lfn is added, e.g., Append "b" will add a record to the file with the lfn "b" whatever is the current file):

all...endall: fastest possible scan but must be used with care if records are to be altered.

append: adds a record to the current or specified while with the current values of the field variables; i.e., it 'clones' the current record of that file.

back: moves back one record.

delete: deletes the current record of the current or specified file.

first: makes the first record of the file the current record.

last: makes the the last record the current record.

next: makes the next record the current record.

When assigning values to variables — i.e., let — lfn's are used thus:

let b.initials\$ = a. initials\$

or (if "a" is the current file)

let b.initials\$ = initials\$

A series of such assignments need then to be followed by:

update "b" (which will amend the current record in "b"), or  
append "b" (which will add a new record to "b")

unless the values assigned to the fields in lfn "b" are required to be held there only temporarily since otherwise they will be lost on moving to another record in lfn "b" or on closing the file. An assignment of a value to a field merely changes the current value of that field without permanently affecting the \_\_dbf file.

I decided that, in my scheme of filenames, files should have the same lfn as their filename which will invariably consist of a single letter. I made provision for the opening of a "main" file for each program which must have a name of more than one character. Having decided this, files can be opened by the following two procedures:

```

proc op;f$
    if len(f$)>1
        open f$

```

```

    else
        open main.wdr$+f$ logical f$
    endif
    endproc

```

```

proc ops
    local no
    let no=1
    while no<len(main.files$)+1
        op;main.files$(no)
        let no=no+1
    endwhile
    endproc

```

It will be noted that these two procedures refer to main.files\$ and main.wdr\$. This is because I decided to provide each program with a "main" \_\_dbf containing the following program variables:

```

wdr$
sdr$
scdr$
budr$
files$
screens$
menus$
levels$) The significance of these two
variables
prints$) will be explained later.

```

I will also be explaining the creation of this \_\_dbf file and others later. The reason the procedures refer to main.wdr\$ and main.files\$ rather than just wdr\$ and files\$ is because once a file has been opened it becomes the current file and so displaces lfn "main" from this position. Without this full variable description ops and op would look to the most recently-opened file for wdr\$ and files\$ and, if it did not contain such fields, global variables would be sought.

## Listings

Until now I have embodied individual listings in the text but from now they will be appended as a listing. In the next part I will explain how listings may be obtained. Suffice it to say at this stage that all procedures start with the word "proc" at the margin and all other lines are indented. Thus if any lines, other than "proc" lines, in the printed listing appear to start at the margin they are the continuation of the previous line. Also hyphens at the end of the previous line of which they are the continuation are not part of the listing.

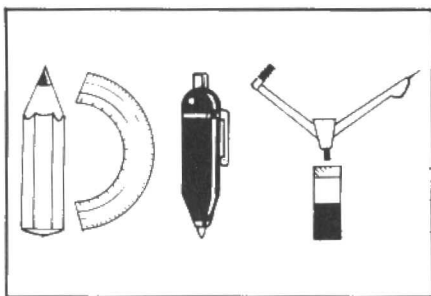
A \_\_dbf file can be created by a series of direct commands:

create "filename" or, if another file is open  
create "filename" logical "lfn"

followed by the names of the fields in the order in which you wish to access them. When all fields have been entered, enter 'endcreate'.

If you discover on debugging that you have made a mistake or you wish to add a field or change their sequence, you have no choice but to 'kill' the file and type it all





again with the attendant risk of error. As a result I prefer to make a program which creates the database for each program. Listing one is of my program "make-comm" will create the database for the program "comm".

So far as makemain is concerned this will create a file and allow the initial values of the system information to be entered. The complete database can be created by entering 'start'.

Individual \_dbf files can be created by calling the specific procedure, e.g., if you enter 'maked' the \_dbf prg\$+"d" will be regenerated. The procedure init adds a single record to each file, empty except that the fields ind1, ind2 and ind3 are set to a high value. The reason will become clear in the next part.

If in program development it proves necessary to change a particular \_dbf file:

a. Close the file concerned

- b. Save the program you are working on
- c. Load the "make" program related to that program
- d. Edit the "make" procedure for the file e.g., for file "a" — makea.
- e. Save the "make" program
- f. Execute it by typing "start >Enter<"
- g. Reload the program on which you were working.

This will replace the file concerned but will, of course, destroy any data in that file so is acceptable only during development before the program has been used. With the next part I will be publishing a listing of my program "reorgdb" which allows fields to be added to or deleted from \_dbf files or the order of fields changed without losing the data.

Since init, makemain and start will always be the same in any of these database creation programs it is worth saving these procedures as the programme "makegen", replacing the other three "make" procedures with:

rem Insert procs to make database files e.g. error makea

An alternative procedure would be to include, during development, init, the make procedures and the make program's "start" procedure — renamed as "make" — in the program being developed. This would simplify the recreation of \_dbf files during development. At the end

the two separate programs could be separated out by the following procedure:-

- a. Save the program as finally edited under its program name, e.g., "comm"
- b. Delete all procedures except init and those beginning with "make"
- c. Save as "make..." (the... representing the program name, e.g., "make comm".)
- d. Reload the original program
- e. Delete init and all procedures beginning with "make"
- f. Save again under the program name (e.g., "make").

Database files once created will be "open". While open they are at risk and if the computer power is interrupted they may be corrupted. At the end of a session they may be closed individually by the command "Close" repeated for every open file. Alternatively the Command "Quit" which finishes the Archive session closes all open files, as does "New" which clears the present program, while leaving Archive still running. In the next part I will include a procedure which closes all open files without losing either Archive or the resident program.

In part three I will deal with the block closing, sorting and reorganisation of database files and general purpose "start" and "end" procedure.

### QL SUPERTOOLKIT II by Tony Tebby THE ULTIMATE QL ENHANCEMENT

Over 118 Commands:— Full Screen Editor, Key Define Print Using, Last Line Recall, Altkey, Job Control, File Handling, Default Directories, Extended Network.

16k Eprom Cartridge Version ..... @£ 24.15d  
Configurable Version on Microdrive ..... @£ 23.00d

### MIRACLE SYSTEMS PRODUCTS

QL Expanderam Ok board, fit your own drams. Offer open only while stocks last ..... £ 23.00d  
QL Trump Card 768k (Toolkit II etc) ..... @£259.90b  
QL Trump Card 512k (Toolkit II etc) ..... @£213.90b  
QL Trump Card 256k (Toolkit II etc) ..... @£149.50b  
QL Expanderam 512k Thru Card ..... £133.40b  
QL Expanderam 256k Thru Card ..... @£ 80.50b  
Drams to suit above 41256/15 ..... @£ 8.05c  
QL Modem ..... @£ 48.30d  
QL Centronics Printer Interface ..... @£ 28.75d  
QL Modaptor ..... @£ 37.95d

### QL HARDWARE

Single 3.5" Disc Drive & (Own PSU) ..... @£ 97.75a  
Dual 3.5" Disc Drive & (Own PSU) ..... @£188.60a  
3.5" POWER REG. The only real solution to your QL overheating (switched mode power supply run cold) ..... @£ 23.00c  
QL Keyboard Membrane ..... @£ 11.50d  
QEP III Advanced Eprom Programmer ..... @£121.90d  
Care Eprom Cartridges each ..... @£ 5.75c  
Eprom 27128 250ns/16k ..... @£ 5.75c  
ULA Chip ZX8301 ..... @£ 15.64c

### MAGNETIC MEDIA

Microdrives (each) ..... @£ 1.98c  
3.5" (each) d/s disc ..... @£ 1.61c  
3.5" (10 of) d/s discs ..... @£ 13.80c

### THE Q CONTROL BOARD

As specified in the Colin Opple Connexions Series — Please call for details ..... @£ 89.95d  
Analogue/detail chip ..... @£ 26.91c  
Cross Assembler 8048 or 6502 ..... @£ 18.40c

### HOW TO ORDER:

ALL PRICES INCLUDE VAT

By Post. Enclose your Cheque/PO made payable to CARE Electronics.  
Or use ACCESS/VISA. Allow 7 days for delivery

### TONY TEBBY SOFTWARE (QJUMP)

OPAC III new from the house of QJump, a totally new version of ORAM and OPAC. Available June 89.

Please phone for further details.

QFLP (Micro/P disc interface upgrade) ..... @£ 14.95d  
QMON II Microdrive ..... @£ 19.95d  
QMED (Medic disc interface upgrade) ..... @£ 14.95d  
QPTR Pointer Interface m/drive ..... @£ 34.50d  
QPTR Pointer Interface + 3.5" disc ..... @£ 29.90d  
QTPY Type/Spell Checker ..... @£ 29.90d

### ZITASOFT SOFTWARE by Steve Jones

LOCKSMITH copies M/DRIVE — M/DRIVE ..... @£ 11.50c  
4MATTER + LOCKSMITH copies M/DRIVE — DISC ..... @£ 23.00c  
SHRIVEL memory shrink prog user definable ie 128k or 192k or 256k etc ..... @£ 6.90c  
TOOLCHEST utilities to allow the creation of customised mdv doctor pron ..... @£ 11.50c

NEW SIDEWINDER print screens or parts of screens from postage stamp size to large banners, print sideways, invert, scale and now comes with colour driver for LC10 or JX80C and accepts desk top publishing files, multiple labelling also included. New printer config supports a large range of printers, even 24pin.

Upgrade to new Sidewinder ..... £ 6.90c  
New Sidewinder ..... £ 23.00c

### MONITORS (Price including lead)

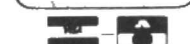
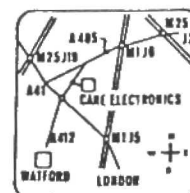
Philips BM7502 Green Hi-Res ..... @£ 89.93a  
Philips CM8833 Colour Med-Res ..... @£271.41a  
Philips AV7300 TV/tuner for above ..... @£ 69.00b

### READYMADE LEADS

RGB QL to Phono ..... @£ 5.75c  
RGB 8-6 pin DIN ..... @£ 7.13c  
RGB 8-7 pin DIN (Hitachi) ..... @£ 7.13c  
RGB 8-7 pin DIN (Ferguson) ..... @£ 7.13c  
RGB 8 pin to SCART (Euro) ..... @£ 11.50c  
6-way PCC 25way 'D' (Printer-Ser 1) ..... @£ 9.89c

**CARE ELECTRONICS**  
OPEN  
9am-5pm Mon-Thu  
9am-4pm Fri-Sat

800 ST ALBANS ROAD,  
GARSTON, WATFORD,  
HERTS. WD2-6NL  
Tel: 0923-672102



**Text<sup>87</sup>**  
**NOW IN STOCK**

Please add carriage  
a=£11.50 b=3.45  
c=£1.38 d=£2.30





# SUPER BASIC

In the final summing-up of the Calculator project, Mike Lloyd adds menu control for a polished finish.

**C**onnections between real-life screen and the pixel world of computer programs are often links by analogy. Thus an icon display becomes a desk-top and a collection of bytes on a storage device becomes a file. A collection of useful utilities is a toolkit and a temporary work space is a scratchpad. Such powerful imagery helps computer users understand what is happening among the chips.

One image which needs to be impressed on all programmers for the benefit of all computer users is that the menus of a computer program are its knobs, levers, switches and dials. This was brought home to me recently while I was testing a word processing package. It had all the mod cons of spelling checker, WYSIWYG drawing facilities and comprehensive printer control. All its functions, however, were buried deep in a gigantic menu hierarchy so that even deleting a character needed five keypresses. Somewhere among the sophistication and complexity of the program the user's needs had been forgotten.

## Corkscrews and plugs

As one software engineer put it, "Corkscrews and electric plugs are easy to use only because a designer has taken the trouble to make them so; programmers seem so engrossed with making programs work that they forget to make them also easy to use."

Even on simple programs like the *QL-Calculator* project it is important to make users feel at home as soon as possible. User confidence can be built by making program control logical and familiar, often by emulating standards set by other software. Confidence is maintained by meticulous attention to error-trapping, by applying consistency in the choice of control keys, by giving users an escape route before committing them to any irrevocable act, and so on.

### Listing 4.1

```
4100 DEFine PROCedure Store_Menu
4105 LOCAL x, k
4115 FOR x = 1 TO 5: Show_Menu x, Store$(x)
4120 Flag "-->"
4125 k = Fetchkey (Fkey$)
4130 IF k
4135   Store$(k) = To_Dec$ (In_Base, Num$)
4140   Show_Menu k, Store$(k)
4145   Flag "F" & k: PAUSE 50
4150 END IF
4155 END DEFine Store_Menu
```

### Listing 4.2

```
4200 DEFine PROCedure Flag           (Text$)
4205 Calc_Wndo: AT 8, 11: Hue 2: PRINT Text$;
4210 END DEFine Flag
```

### Listing 4.3

```
4300 DEFine PROCedure Fetch_Menu
4305 LOCAL x, Key
4315 FOR x = 1 TO 5
4320   IF ABS (Store$(x)) < In_Base^8
4325     Show_Menu x, Store$(x)
4330   ELSE
4335     Show_Menu x, " * * *": Fkey$(x + 1) = "?"
4336   END IF
4337 END FOR x
4340 Flag "<-"
4345 Key = Fetchkey (Fkey$)
4350 IF Key
4355   Num$ = From_Dec$(In_Base, Store$(Key))
4360   Flag "F" & Key
4365 END IF
4370 END DEFine Fetch_Menu
```



Readers who have followed the development of the QL-Calculator program will know that error-trapping has been achieved in part by specifying a string of valid keypresses and responding only to them. When sub-menus are displayed the range of valid keys is reduced to six, the five function keys and the escape key, which provides a "hidden option" in each menu. Knowing that the Fetchkey function, published earlier, will respond only to those keys makes further error-trapping in this month's modules much more straightforward.

### Using the function keys

Although designing a menu structure using only the function keys would impose severe restrictions on a large program it is an ideal way to maintain menu consistency in a small program. Three of the sub-menus are connected with data storage and retrieval; the fourth controls system options and the fifth allows the program to be exited.

Listing 4.1 controls the storage of a value currently appearing on the input line. The menu displays any values which have been saved previously so that the user can avoid over-writing them accidentally. A prominent arrow pointing from the input line towards the menu area indicates the direction of traffic. A user presses a function key to store the input line value. If he changes his mind after selecting this sub-menu he can press the escape key to return to the main input mode.

### All values in decimal

The QL-Calculator stores all values in decimal, no matter what the input or output base might be, hence the call to the conversion routine at line 4135. When a value is saved the menu display is updated and the function key pressed is displayed momentarily near to the input line.

The "flag" routine at listing 4.2 displays arrows and function key names next to the input line. This short procedure saves five repetitions of the code it contains. The next listing is the opposite of the first, retrieving values which have been saved previously. Once again users benefit from an informative sub-menu display showing the values which have been saved. The arrow now points from the menu towards the input line but the basic principle of pressing a function key to obtain a result remains the same.

For obvious reasons values which are too great to be displayed in the current input base cannot be selected. They are replaced in the menu display by a line of

#### Listing 4.4

```
4400 DEFine PROCedure Constant_Menu
4405 LOCAL x, Key, Number(5), Temp$(10)
4415 FOR x = 1 TO 5
4420 IF ABS (Const$(x)) < In_Base^8
4425   Number(x) = Const$(x)
4430   Temp$ = Const$(x, " " INSTR Const$(x) +1 TO)
4435   Show_Menu x, Temp$
4440 ELSE
4445   Show_Menu x, " * * *": Fkey$ (x +1) = "?"
4450 END IF
4455 END FOR x
4460 Flag "<-"
4465 Key = Fetchkey (Fkey$)
4470 IF Key
4480   Num$ = From_Dec$(In_Base, Number (Key))
4485   Flag 'F' & Key
4490 END IF
4495 END DEFine Constant_Menu
```

#### Listing 4.5

```
4500 DEFine PROCedure Set_Menu
4502 LOCAL x, Loop, Key
4506 REPEAT Loop
4508   Show_Menu 1, "IN = " & Base$ (In_Val)
4510   Show_Menu 2, "OUT = " & Base$ (Out_Val)
4512   Show_Menu 3, "PRT = " & Prt$ (PrtOn +1)
4514   Show_Menu 4, "MODE= " & Mode$ (IntOnly +1)
4516   Show_Menu 5, "RETURN"
4518   Key = Fetchkey (Fkey$)
4520   SElect ON Key
4522     = 0, 5: EXIT Loop
4524     = 1: In_Val = 1 +In_Val MOD 4
4526     = 2: Out_Val = 1 +Out_Val MOD 4
4528     = 3: Set_Printer
4530     = 4: IF IntOnly
4532         IntOnly = 0
4534     ELSE
4536         IF In_Val = 1 AND Out_Val = 1
4538             IntOnly = 1
4540         ELSE
4542             Warning
4544         END IF
4546     END IF
4548   END SElect
4550   IF In_Val > 1 OR Out_Val > 1: IntOnly = 1
4552   In_Base = CODE (BaseVal$ (In_Val))
4554   Out_Base = CODE (BaseVal$ (Out_Val))
4556 END REPEAT Loop
4558 END DEFine Set_Menu
```

#### Listing 4.6

```
4600 DEFine PROCedure Set_Printer
4605 IF PrtOn
4610   PRINT#5; CHR$(7): CLOSE#5
4612   PrtOn = 0
4615 ELSE
4620   OPEN#5, ser1: PRINT#5; CHR$(7);
4625   PrtOn = 1
4630 END IF
4635 END DEFine
```



asterisks and the appropriate menu key value in the *Valid\$* string is replaced by a question-mark, making it unselectable, to preserve the integrity of the error-trapping system.

Listing 4.4 is very similar to its predecessor, except that it uses values stored in the "constants" array. Unusually, they are stored together with their labels in a single character array. Coercion is used to extract the values, while the INSTR function is used to strip out the textual descriptions. As previously, values too great for the current input base are suppressed.

The third sub-menu is used to set what might be termed the program "system variables". The most immediately useful are those controlling the input and output bases. Pressing either F1 or F2 cycles round the available options until the desired ones are reached. This is an unwieldy way of presenting a large number of options but it is manageable here.

## Integers or real numbers

Values represented in binary, octal or hexadecimal are always integers, while decimal values can also be real numbers. Users can toggle between integer and real number input by pressing F3.

The F4 key toggles the printer option on and off using the controlling code contained in the next procedure definition. The F5 key, which otherwise would be redundant, is linked to the ESC key to give an overt means of leaving this menu when all choices are complete. In this program users can make many selections in this sub-menu before opting to return to the main input mode.

After each keypress the module must update the menu display and check whether it must force the QL-Calculator into integer mode because a non-decimal base has been chosen. Similarly, it must prevent the user selecting the real number mode when a non-decimal base is present. This has been achieved by reprinting the menu after every keystroke, a cumbersome solution but one requiring very little code.

## Nest structures and quit

Listing 4.6 opens or closes a channel to the printer via the *ser1* interface and uses Epson control codes to make the printer beep an acknowledgement. The variable "PrtOn", which is set or re-set here, is referred to elsewhere in the program to determine whether or not hard copy is required.

The *Quit* sub-menu is unusual in that it is programmed as a function which returns a value to the main input mode. This

### Listing 4.7

```
4700 DEFine FuNction Quit_Menu
4705 LOCAL x, Loop, Key
4715 Show_Menu 1, "Quit": Show_Menu 2, "Resume"
4720 FOR x = 3, 4, 5: Show_Menu x, " "
4725 Key = Fetchkey (Fkey$)
4730 IF Key = 1: RETURN 1
4735 RETURN 0
4740 END REPEAT Loop
4745 END DEFine Quit_Menu
```

### Listing 4.8

```
4800 DEFine PROCedure Quit_Calc
4805 PrtOut "*" END CALC"
4810 IF PrtOn: CLOSE#5
4815 END DEFine Quit_Calc
```

arrangement is for the benefit of the purists who like to ensure that all nested structures are exited properly prior to quitting a program. Just two options are displayed, allowing users to confirm the decision to quit or to resume using the calculator. If the exit option is confirmed the final listing, 4.8, tidies by printing an explanatory message and closing the printer channel.

Utilities such as QL-Calculator are most useful, as executable jobs can be multi-tasked. This can be achieved only by machine code programs or ones which have been compiled with one of the many proprietary compilers available for the QL. The following comments are based on the Digital Precision Turbo compiler, the most powerful but also the most demanding compiler on the market.

By compiling the QL-Calculator it can be loaded much faster; it can have non-destructive windows which return the screen display to exactly what it was before the program was accessed; its error-trapping can be much improved and more significant digits can be displayed.

The compiled version is likely to be larger in size; routines to turn the program on and to suspend SuperBasic would need to be added, and the increase in speed for which compilers are noted would largely be wasted on such an interactive program.

Although QL-Calculator will compile with Turbo with little difficulty, some re-writing is necessary to take full advantage of the facilities the compiler offers. The compiled program needs no special window-handling or hotkey arrangements if Qram is used because it can be accessed and removed by pressing CTRL-C. In the absence of Qram, procedures similar to those included with Turbo will need to be incorporated to restore screen displays and turn the calculator on and off.

The SuperBasic version of the program cannot detect whether it has been successful in opening a channel to the printer. The printer might be busy or the channel might be allocated for another purpose. The Turbo toolkit includes a new keyword, *Device\_status*, which can detect such problems before they cause a crash.

## How to hold the digits

The Turbo nine-digit precision is a different matter, because it affects the QL-Calculator conversion routines adversely. A simple cure is to re-write these routines so that they do not depend on the presence of the "E" symbol to trigger overflow action. The additional digits by extending the calculator display area, increasing the size of appropriate strings by two character spaces and amending the code accordingly. These changes affect a few modules in the program and should be undertaken with great care, keeping an unmodified version of the program in a safe place until the changes have been tested.

The easiest way of obtaining a compiled version of QL-Calculator is to order one from Microdrive Exchange when it appears there. The program has been partly re-written and extended to include a larger variety of useful constants, more menu options and a useful ASCII code display.

● *The next issue of SuperBasic answers many of your queries about technical aspects of SuperBasic. If you have found a problem with SuperBasic programming, or you would like to share a programming tip, write to Mike Lloyd, Sinclair QL World, Focus Magazines Ltd, Greencoat House, Francis Street, London SW1 1DG.*



**THE**

# P+R:O=G<S

If you have a program worthy of consideration, send it to 'The Progs',  
Sinclair QL World, Greencoat House, Francis Street, London SW1P 1DG.  
We pay for everything published at the usual page rates.

## Program of the month

### CUBE by Dirk de Mal

**C**ube is the game in which you unscramble the colours in the cube in as few turns as possible and in the correct colour order.

The game is very simple to play. There are five levels of difficulty from 1 (difficult) to 5 (easy).

The rest is self-explanatory.

```
10 MODE 8:CLS#1
20 BORDER 0:WINDOW 450,220,35,15:PAPER#2,0:CLS#2
30 PAPER 0:CLS:CLS#0:screen
40 inkt:was:instruct:TIME:number
50 jumble
60 DEFINE PROCEDURE screen
70 BORDER 0:CSIZE 0,0:AT 15,5:PRINT
  " november 1987 DIRK DE MAL"
80 a1=1:a2=6:a3=2:b1=1:b2=6:b3=2:c1
  =1:c2=6:c3=2:pva=b1:pvb=b2:pvc=b3:pv
  d=a2:pve=b2:pvf=c2
90 BORDER 0
100 CSIZE 3,1:FOR R=1 TO 80:CURSOR
  190,R:INK RND(1 TO 100):PRINT "CUBE
  ":NEXT R:begin:begin:CLS:UNDER 1:AT
  0,11:INK 6:PRINT "CUBE":UNDER 0:nu
  mber
110 OPEN#3,scr_50x40a150x50
120 OPEN#4,scr_50x40a202x50
130 OPEN#5,scr_50x40a254x50
140 OPEN#6,scr_50x40a150x92
150 OPEN#7,scr_50x40a202x92
160 OPEN#8,scr_50x40a254x92
170 OPEN#9,scr_50x40a150x134
180 OPEN#10,scr_50x40a202x134
190 OPEN#11,scr_50x40a254x134
200 END DEFINE
210 DEFINE PROCEDURE jumble
220 icheck
230 END DEFINE
240 DEFINE PROCEDURE inkt
250 PAPER#3,a1:PAPER#4,b1:PAPER#5,c
  1
260 PAPER#6,a2:PAPER#7,b2:PAPER#8,c
  2
270 PAPER#9,a3:PAPER#10,b3:PAPER#11
  ,c3
280 END DEFINE
290 DEFINE PROCEDURE was
300 CLS#3:CLS#4:CLS#5:CLS#6:CLS#7:C
  LS#8:CLS#9:CLS#10:CLS#11
310 END DEFINE
320 DEFINE PROCEDURE looka
```

```
330 LET b1=a1:inkt:pann 4:a1=c1:ink
  t:pann 3:LET c1=pva:inkt:pann 5:var
340 END DEFINE
350 DEFINE PROCEDURE lookb
360 LET b2=a2:inkt:pann 7:a2=c2:ink
  t:pann 6:LET c2=pvb:inkt:pann 8:var
370 END DEFINE
380 DEFINE PROCEDURE lookc
390 LET b3=a3:inkt:pann 10:a3=c3:in
  kt:pann 9:LET c3=pvc:inkt:pann 11:v
  ar
400 END DEFINE
410 DEFINE PROCEDURE lookd
420 LET a2=a1:inkt:scrol 6:a1=a3:in
  kt:scrol 3:LET a3=pvd:inkt:scrol 9:
  var
430 END DEFINE
440 DEFINE PROCEDURE looke
450 LET b2=b1:inkt:scrol 7:b1=b3:in
  kt:scrol 4:LET b3=pve:inkt:scrol 10
  :var
460 END DEFINE
470 DEFINE PROCEDURE lookf
480 LET c2=c1:inkt:scrol 8:c1=c3:in
  kt:scrol 5:LET c3=pvf:inkt:scrol 11
  :var
490 END DEFINE
500 DEFINE PROCEDURE var
510 pva=b1:pvb=b2:pvd=a2:pve=b2:pvc
  =b3:pvf=c2
520 END DEFINE
530 DEFINE PROCEDURE TIME
540 CSIZE 0,0:AT 17,0:INK 0:PRINT F
  ILL$(" ",50):INK 6:AT 18,9:PRINT "1
  =HARD 5=EASY":AT 17,0:INK 3:PRINT
  "Which level are you going to play?
  ":recheck
550 END DEFINE
560 DEFINE PROCEDURE recheck
570 a=CODE(INKEY$(-1))
580 REPEAT loop
590 SELECT ON a
600 =49:K=8
610 =50:K=9
620 =51:K=10
630 =52:K=15
640 =53:K=20
650 =REMAINDER :negsound:recheck
660 END SELECT
670 CSIZE 0,0:AT 18,9:PRINT FILL$("
  ",14)
```



```

680 Y=0:sprint:mix:info:beurt
690 END DEFine
700 DEFine PROCEDURE beurt
710 CSIZE 0,0:AT 17,0:PRINT FILL$(" "
  ",30):AT 17,6:INK 2:PRINT "You sti
  ll have ";K;" turns."
720 LET m=K-1
730 K=m
740 check
750 IF m=-1 THEN INK 6:CLS:CSIZE 3,
  1:AT 4,7:PRINT "YOU FAILED!!!":blow
  up:wrong
760 jumble
770 END DEFine
780 DEFine PROCEDURE check
790 IF a1=1 AND a2=6 AND a3=2 AND b
  1=1 AND b2=6 AND b3=2 AND c1=1 AND
  c2=6 AND c3=2 THEN victory:GO TO 10
  70
800 END DEFine
810 DEFine PROCEDURE victory
820 RESTORE 900
830 READ u
840 IF u=0 THEN RETURN
850 READ D
860 BEEP 7000*D,u
870 PAUSE 3
880 IF BEEPING THEN GO TO 880
890 GO TO 830
900 DATA 104,1,104,1,91,1,109,1.5,1
  04,.5,91,1,81,1,81,1,76,1,81,1.5,91
  ,.5,104,1,91,1,104,1,109,1,104,1
910 DATA 104,.5,91,.5,81,.5,76,.5,6
  5,1,65,1,65,1,65,1.5,76,.5,81,1,76,
  1,76,1,76,1,76,1.5,81,.5,91,1
920 DATA 81,1,76,.5,81,.5,91,.5,104
  ,.5,81,1.5,76,.5,66,1,58,.5,76,.5,8
  1,1,91,1,104,2,104,.5,104,.5,0
930 END DEFine
940 DEFine PROCEDURE mix
950 CSIZE 0,0:INK 6:AT 17,0:PRINT F
  ILL$(" " ",50):AT 17,10:PRINT "Now ju
  mbling."
960 FOR R=1 TO 20
970 t=RND(1 TO 6)
980 IF t=1 THEN looka
990 IF t=2 THEN lookb
1000 IF t=3 THEN lookc
1010 IF t=4 THEN lookd
1020 IF t=5 THEN looke
1030 IF t=6 THEN lookf
1040 NEXT R
1050 END DEFine
1060 DEFine PROCEDURE wrong
1070 CSIZE 0,0:AT 18,2:AT 18,3:INK
  2:PRINT "Do you want to try again?
  (y/n)"
1080 a=CODE(INKEY$(-1))
1090 Y=2
1100 REPEAT loop
1110 SELECT ON a
1120 =89:sprint:RUN
1130 =78:sprint:einde
1140 =121:sprint:RUN
1150 =110:sprint:einde
1160 =REMAINDER :negsound:wrong
1170 DEFine PROCEDURE einde
1180 INK 4:BORDER 0:CLS:CLS#0:CSIZE
  3,1:AT 5,9:PRINT "BYE!!!":FOR a=1
  TO 4:begin:NEXT a:CLS:STOP
1190 END DEFine
1200 DEFine PROCEDURE begin
1210 RESTORE 1300
1220 FOR I=1 TO 54
1230 READ N,D
1240 IF D=1.3 THEN D=1.7
1250 PAUSE .5
1260 BEEP 3000*D,N
1270 IF BEEPING THEN GO TO 1270
1280 BORDER D,RND(6)

```

```

1290 NEXT I
1300 DATA 51,.5,55,.5,51,1.3,111,1,
  3,111,1.3,71,.5,82,.5,87,.5,71,.5,5
  1,.5,55,.5,51,.5,38,.5,44,.5,51,.5
1310 DATA 44,1.3,99,1.3,99,1.3,99,.
  5,111,.5,121,.5,99,.5,71,.5,76,.5,7
  1,1.3,62,.5,55,.5
1320 DATA 51,.5,55,.5,62,.5,71,.5,6
  2,.5,71,.5,82,.5,87,.5,82,.5,87,.5,
  99,.5,111,.5,111,.5,121,.5,133,.5,1
  49,.5
1330 DATA 133,.5,111,.5,121,.5,99,.
  5,111,.5,87,.5,99,.5,82,.5,87,1.3,1
  1,1.3,111,1.5,0
1340 BORDER 0
1350 END DEFine
1360 DEFine PROCEDURE number
1370 INK 6:CSIZE 1,0:CURSOR 95,43:P
  RINT 1
1380 CURSOR 95,86:PRINT 2
1390 CURSOR 95,128:PRINT 3
1400 CURSOR 135,20:PRINT 4:CURSOR 1
  87,20:PRINT 5:CURSOR 237,20:PRINT 6
1410 END DEFine
1420 DEFine PROCEDURE instruct
1430 AT 17,0:PRINT FILL$(" " ",50):AT
  17,0:PRINT "This is how the cube m
  ust be":PAUSE 200
1440 END DEFine
1450 DEFine PROCEDURE pann (x)
1460 FOR a=1 TO 10:PAN#x,6:NEXT a
1470 END DEFine
1480 DEFine PROCEDURE scrol (x)
1490 FOR a=1 TO 10:SCROLL#x,4:NEXT
  a
1500 END DEFine
1510 DEFine PROCEDURE sprint
1520 INK RND(1 TO 6):
1530 IF Y=0:BEEP 100,1:CSIZE 0,0:AT
  17,35:PRINT-CHR$(a):PAUSE 50
1540 IF Y=1:CSIZE 3,1:INK 5:AT 4,20
  :PRINT CHR$(a)
1550 IF Y=2:CSIZE 0,0:AT 18,34:PRIN
  T CHR$(a):PAUSE 50
1560 END DEFine
1570 DEFine PROCEDURE icheck
1580 AT 18,4:INK 1:PRINT "Please ty
  pe numbers 1 to 6."
1590 Y=1:a=CODE(INKEY$(-1))
1600 REPEAT loop
1610 SELECT ON a
1620 =49:BEEP 5000,5:sprint:looka
1630 =50:BEEP 5000,5:sprint:lookb
1640 =51:BEEP 5000,5:sprint:lookc
1650 =52:BEEP 5000,5:sprint:lookd
1660 =53:BEEP 5000,5:sprint:looke
1670 =54:BEEP 5000,5:sprint:lookf
1680 =REMAINDER :negsound:icheck
1690 END SELECT
1700 beurt:jumble
1710 END DEFine
1720 DEFine PROCEDURE negsound
1730 BEEP 100,50
1740 END DEFine
1750 DEFine PROCEDURE blowup
1760 WINDOW 250,100,130,70
1770 FOR a=1 TO 80:SCROLL 4:SCROLL
  -4:BEEP 10000,RND(900 TO 1000),100,
  40,50,30,1,30:NEXT a
1780 WINDOW 450,220,35,15
1790 END DEFine
1800 DEFine PROCEDURE info
1810 OPEN#20,scr_50x31a35x20
1820 FOR x=1 TO 4:PAPER#20,2:SCROLL
  #20,x:NEXT x
1830 FOR x=1 TO 4:PAPER#20,6:SCROLL
  #20,x:NEXT x
1840 FOR x=1 TO 4:PAPER#20,1:SCROLL
  #20,x:NEXT x
1850 END DEFine

```



## Printer Spooler

Ian Jackson presents a useful small utility to allow the QL to print documents while carrying-out other tasks.

This multi-tasking printer spooler hex loader program should first be typed-in and run. Then, the spooler can be started — for multi-tasking operation — using:

EXEC\_\_W mdv1\_\_spooler

To switch the cursor between Basic and the spooler, press CTL—C.

To customise the spooler for your printer, type-in the install program and alter it to suit your printer. Then run it and the copy of the spooler in mdv1\_\_ will be altered. To save the install program, type:

SAVE\_\_ME

The data in the install program starts with the y-coordinate at the of the display window, for mode 4 and 8. Then follows the preamble sequence, which is sent to the printer before printing starts. The printer options which you select in the program are defined from then onwards.

At the end of each definition of the preamble or a printer option you should put 'end', as you should after all your printer options have been defined. Within each definition list you may use 'esc' to send CHR\$(27) to the printer, 'If' to send CHR\$(10), or you may put '...' to send the character ..., or just a number to send that character number.

```
100 REMark hex loader for Printer Spooler
110 :
120 ad=respr(1760): a=ad: hl$='0123456789ABCDEF':
errflag=0: RESTORE
130 FOR i=1 TO 110
140 READ d$,check: sum=0
150 FOR j=1 TO 16: b=16*(d$(j*2-1) INSTR hl$)+(d$(j*2) INSTR hl$)-17: POKE a,b: a=a+1: sum=sum+j*b
160 CLS#0: PRINT#0,i: IF sum<>check THEN PRINT 'Error in line ' ;190+10*i: errflag=1
170 END FOR i
180 IF NOT errflag THEN SEXEC mdv1__spooler,ad,1756
.536
190 :
200 DATA '601003BA00004AFB000753706F6F6C65',11967
210 DATA '7200584FDFA065C700172FF41FA03B6',15991
220 DATA '4E424A086600024E2C887027720076FF',12880
230 DATA '4E43702872004E43702972074E436100',8523
240 DATA '029443FA0396347800D04E9261000320',10246
250 DATA '43FA03A04E92610003366100027843FA',11506
260 DATA '03B6347800D04E92610003047002343C',8143
270 DATA '0029363CFFFF43FA06384E43A80670C',12525
280 DATA '0C0000FB660270EB600002A653416700',10614
290 DATA '01C643FA061A3281700172FF76012049',11694
300 DATA '4E424A08670C0C0000F96700017E6000',8455
310 DATA '02802D48000441FA056270014E424A80',10525
320 DATA '670C0C0000F7660270EA600002542D0',9808
330 DATA '00086100020043FA047E347800D04E92',12512
340 DATA '6100028C700243FA05C8343C00294E43',9857
350 DATA '4A90670C0C0000FB660270EB60000222',9567
360 DATA '534143FA05AA3281206E0008700743FA',11884
370 DATA '0468343C0028363CFFFF4E43A806600',12497
380 DATA '01F449FA058A3E1C53476B38101C0200',7155
390 DATA '00DF4BFA046CB0156710DAFC000C4A2D',12246
400 DATA '000166F270F1600001CC70077400142D',9847
410 DATA '0001363CFFFF43ED00024E43A806600',11540
420 DATA '01B460C4610000B449FA04FE51D47011',15580
430 DATA '47FA05344E410201000C0C01000C673E',4565
```

```
440 DATA '7002343C01AA363CFFFF206E000443FA',13973
450 DATA '051E4E434A8067064BFAFFD461387007',14835
460 DATA '3401363CFFFF206E000843FA05024E43',10517
470 DATA '4A8067064BFAFF8611C4A1467B06100',13644
480 DATA '011443FA045C347800D04E92610001C0',12176
490 DATA '7000600001400C0000F657D4674A2056',11065
500 DATA '610000B261000188347800D043FA0454',12328
510 DATA '4E92700E4E4370014E431E01700F4E43',7968
520 DATA '020700DF0C070043670A0C07004166E2',8845
530 DATA '588F60AA6104584F4ED5610000B83478',12274
540 DATA '00D043FA045A4E924E75610000A843FA',14187
550 DATA '0250347800D04E9243FA0474347800D0',13983
560 DATA '4E92347800D043FA01F64E92363C00C8',14544
570 DATA '61306000FE06617C347800D043FA01E4',15656
580 DATA '4E926100010A43FA01F64E9261000120',10062
590 DATA '760060082600347800CC4E92700572FF',13918
600 DATA '4E4148E7C0C0700893C972FF4E414CDF',18587
610 DATA '03034E7548E7F8F0613A3800444E344',14511
620 DATA '47FA0206383340FE71043F340003478',11939
630 DATA '00D04E924CDF0F1F4E75347800CC4880',13173
640 DATA '4E92701172174E43347800D043FA036E',12904
650 DATA '4E9260E048E7F042056701072FF7AFF',18936
660 DATA '4E414A01660643FA00DC600443FA00DE',14676
670 DATA '700D720274014E4370204E43701072FF',12616
680 DATA '74FF4E414A016606701172024E434CDF',11374
690 DATA '020F4E752F007002206E00084E42201F',6167
700 DATA '2F007002206E00044E42201F6700F02C',9167
710 DATA '6100FF62701072FF7AFF4E414A016704',12565
720 DATA '720A60027220205670054E43347800D0',10508
730 DATA '43FA013E4E92700E4E4370014E430C01',7744
740 DATA '00E866F67005720A4E436000FCEE48E7',17034
750 DATA 'F050701072FF7AFF4E414A01670A2056',11265
760 DATA '7005720A76FF4E434CDF0A0F4E757E07',11564
770 DATA '7C020C06000666027C0743FA02CF7006',10577
780 DATA '12C651C8FFFC702643FA02C04E435446',15093
790 DATA '0C0600076DDC51CFFF084E7501E40167',15725
800 DATA '000A00100200000C0000000002000016',568
810 DATA '0000001E0003636FE00001620205072',6861
820 DATA '696E7465722053706F6F6C6572202020',10747
830 DATA '202000222020202020436F7079726967',10979
840 DATA '68742028432931393852049616E204A',8699
850 DATA '61636B736F6E00255479706520746865',12213
860 DATA '2066696C656E16D652C206F7220454E',10945
870 DATA '54455220746F2073746F703A20000002',7516
880 DATA '222E00194279652066726F6D20707269',12206
890 DATA '6E7465722053706F6F6C65723B00001F',9252
900 DATA '20707265737320636F6E74726F6C2043',12533
910 DATA '20666F72206F74686572204A6F622E00',10289
920 DATA '000E49206361E27742066696E642022',10134
930 DATA '0016507265737320463120746F20636F',11012
940 DATA '6E74696E75652E2000000000000000',2866
950 DATA '002C00000042000005C0000000000000',1404
960 DATA '000000000006A0000000000000000000',636
970 DATA '0088009C001441206C96E6520776173',11548
980 DATA '20746F6F206C6F6E72E0017546F6F20',9992
990 DATA '6D616E79206368616E6E656C73206F70',13272
1000 DATA '656E210000046696C6520696E207573',10889
1010 DATA '652E001C596F7572206F7074696F6E73',13578
1020 DATA '206172652053494C42454446502E200',8297
1030 DATA '0012496E707574206725666665722066',12444
1040 DATA '756C6C2E000F5072696E74657220696E',12201
1050 DATA '207573652E000020456E746572207468',11044
1060 DATA '65207072696E746572206D6F6465206F',12609
1070 DATA '7074696F6E733A202020202020202020',6205
1080 DATA '20202020202020202020202020202020',4352
1090 DATA '20202020202020202020202020202020',4484
1100 DATA '53010F000000000000000000004021B34',2344
1110 DATA '000000000000000000004031B570100000',2068
1120 DATA '0000000042021B450000000000000000',1083
1130 DATA '45021B4D000000000000000000004021B47',2915
1140 DATA '0000000000000000000040601B6C0F1B5141',5111
1150 DATA '00000000050061B43461B4E050000000',2979
1160 DATA '00000000000000000000000000000000',0
1170 DATA '0000000000000000000000000073657231',5451
1180 DATA '00194920686176652066696E69736865',13067
1190 DATA '64207072696E74696E672E000004202D',7575
1200 DATA '20200024205072657373204120746F20',10020
1210 DATA '61626F7274206F72204320746F20636F',11397
1220 DATA '6E74696E75652E200000000000000000',2866
1230 DATA '00000000000000000000000000000025',592
1240 DATA '205072696E74696E67202D2070726573',12348
1250 DATA '732045534320616E6420BE20746F2061',11592
1260 DATA '626F72742E0009010000000001020000',1468
1270 DATA '00000000000000000000000000000000',0
1280 DATA '00000000000000000000000000000000',0
1290 DATA '00000000000000000000000000000000',0
```



# PRINTER SPOOLER

```

100 REMark Spooler install - Brother M1009
110 REMark REM's may be omitted
120 :
130 DATA 0,30:
    REMark top of window y-coordinates for mode 4, 8
140 DATA 'esc','"@','esc','"R',0,'lf','end':
    REMark preamble
150 DATA 'S',15,'end':
    REMark small
160 DATA 'I','esc','"4','end':
    REMark italics
170 DATA 'L','esc','"W',1,'end':
    REMark large
180 DATA 'B','esc','"E','end':
    REMark bold
190 DATA 'E','esc','"M','end':
    REMark elite
200 DATA 'D','esc','"G','end':
    REMark doublestrike
210 DATA 'F','esc','"1',15,'esc','"Q',65,'end':
    REMark fifty chars across
220 DATA 'P','esc','"C',70,'esc','"N',5,'end':
    REMark paged, perforation skip
230 DATA 'end'
240 :
250 DIM cd%(9): RESTORE : READ y4,y8: ad=RESPR(200
0)
260 LBYTES mdv1_spooler,ad
270 adt=ad+2+PEEK_W(ad+2): adf=adt+PEEK_W(adt): ad
l=adt+2+PEEK_W(adf+2)

```

```

280 POKE_W adt+4+PEEK_W(adf+4),y4: POKE_W adt+6+PE
EK_W(adf+6),y8
290 FOR i=0 TO 9: rdy: cd%(i)=v: IF v=-1: EXIT i
300 j=i-(v=-1): FOR i=0 TO 8-j: POKE adf-10+i,32
310 FOR i=9-j TO 9: POKE adf-10+i,cd%(i-9+j)
320 FOR y=0 TO 9
330 adh=adf+y*12: READ i$: v=CODE(i$)
340 IF i$='end' THEN v=-1: EXIT y
350 POKE adh,v: POKE adf+y,v: FOR i=0 TO 9: rdy:
IF v=-1: EXIT i: ELSE POKE adh+i+2,v
360 POKE adh+i,i+(v<-1)
370 END FOR y
380 i=y+(v<-1): POKE adf+i,46: POKE adf+i+12,0
390 FOR j=i+1 TO 9: POKE adf+j,32
400 DELETE mdv1_spooler: SEXEC mdv1_spooler,ad,175
6,536
410 :
420 DEFine PROCedure rdy
430 READ i$
440 IF i$(1)=" " THEN v=CODE(i$(2)): RETURN
450 IF i$='end' THEN v=-1: RETURN
460 IF i$='lf' THEN v=10: RETURN
470 IF i$='esc' THEN v=27: RETURN
480 v=i$: RETURN
490 END DEFine
500 :
510 DEFine PROCedure save_me
520 DELETE mdv1_spoolinst1: SAVE mdv1_spoolinst1
530 DELETE mdv2_spoolinst1: SAVE mdv2_spoolinst1
540 END DEFine

```

## THOR XVI Version 4

Version 4 of THOR XVI is a completely redesigned hardware and software version of the well-known CST THOR XVI, now a 100% Danish production. It is produced by one of the largest Scandinavian manufacturers of high quality measuring instruments vouching for the high standard of our product.

The brand new version of ARGOS (6.40/1.07) introduces a number of new facilities and a much improved full SCSI hard disc handling. ARGOS 6.40 now supports the standard IBM AT keyboards, IBM AT extended and PS/2 extended keyboards, covering 11 different national languages.

Now it is possible to use the advanced facilities of any printer, including laserprinters, thanks to new user defined extended translation tables.

ARGOS windiwing facilities are extended to support separate screen MODE for each job.

In addition to PSION Xchange we deliver free of charge 2 discs with easy to use menu system and utilities.

Present THOR XVI users please register with us in order to request a ROM/Software upgrade. Technical support is already available in the U.K. through P.M. Engineering.

Please request the new 12 pages THOR XVI brochure and price list containing also a description of the new coming products. We are pleased to announce 2 new products:

★ **ARCTURUS EDITOR (Arced)**, the indispensable companion for programmers (also running on THOR 8 and Sinclair QL), introductory price: **GBP £45.00**. Please request a technical description.

★ **C++ COMPILER** for the THOR and QL range of products, is under development.

*Dansoft offers software service contracts for existing THOR customers*

## DANSOFT

*Sole Agent for:*

**THOR INTERNATIONAL COMPUTER SYSTEMS I/S**

Raadhustraede 4 b, 4.sal, DK 1466 Copenhagen K  
Mail to: P.O. Box 59, DK 1002 Copenhagen K, Denmark.  
Phone no: +45 1 930305 (after May 15th: +45 33930305)  
Fax no: +45 1 938292 (after May 15th: +45 33 938292)

## ☆ sinclair QL ☆

### EEC LTD SUPPLIES OF Sinclair QL COMPUTERS

#### QLs FROM £65, NEW PRICE LIST

**BACKUP QL JM ROM.** The QL only. Assembled & tested not S/H keep it as insurance, or for networking. Cheaper than changing to another system.....£65

**BACKUP QL JS ROM.** As above.....£80

**QLs COMPLETE.** Fully tested and with 3 months warranty.

**QL JM ROM.** TV lead, QL Software 2.35, also complete bound user guide for QL, Superbasic, & The Software; Quill-Word Processor, Abacus-Spread Sheet, Archive - for records, Easel-business graphics.....£120

**QL JS ROM.** As above.....£135

#### SPARES & ACCESSORIES When Available.

**QL Software 2.35** .....£15 (or Upgrade. Send Pack/Disk, S.A.E. & £5.)

**Complete bound user guide**.....£15 **QL power supply unit**.....£22

**Microdrive units** .....£20 **JM Roms** .....£10

**Working P/C Boards JM** .....£45 **Joysticks**.....£13

**QL ICs** .....P.O.A.

**LOW COST PRINTERS.** Complete with installation, instructions for Quill, Abacus & Archive.

**SEIKOSHA GP100A.** Centronics. 80 Col, 50 cps, Graphics. Tractor feed. Especially suitable as a second printer for listing, & printouts.....£80

**SEIKOSHA GP550A.** Centronics. 80 Col, 50 cps, NLQ 25 cps. 139 Characters. Graphics. Built-in friction/tractor feed. Prints-underline, italic, expanded, condensed, elite, NLQ, proportional, sub & superscript.....£120

#### QL SOFTWARE CLEARANCE

**Integrated Accounts. Cash Trader. Decision Maker. Home Finance. Entrepreneur. Macro Assembler. Toolkit** .....£18

**Touch & Go (Typing Tutor). Gardener. QL Paint. Reversi. QL Games Cartridge** .....£9

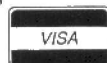
#### TERMS CWO, ACCESS or VISA.

Minimum order £9. P&P £6.00 for printers & QL, other items £3.00. Overseas Enquire.

Fax No: 0753 887149

**EEC LTD**

18-21 Misbourne House, Chiltern Hill, Chalfont St Peter, Bucks, SL9 9UE. Tel: 0753 888866.





## MULTIPLICATION TABLES by Ron Allpress

```

170 REMark ***      PROGRAM IS COMPLETELY      ***
185 REMark ***      ONLY RESET/BREAK WILL STOP IT      ***
620 AT 5,INT((28-LEN(title$))/2):PRINT title$:PAUSE 90
621 CLS
622 IF title$="Multiplication Tables" THEN
623 AT 3,4:PRINT "What is your name?"
624 AT 5,4:INPUT name$
625 AT 7,4:PRINT "Hello, "
626 AT 8,4:PRINT name$;"!":PAUSE 100
627 END IF
701 IF test=1 THEN
702 INK 2:AT 0,1:PRINT"Press <ENTER> if your name is not
703 AT 1,INT((37-LEN(name$))/2):PRINT name$:INK 1
704 END IF
158 REMark ***      Raspberry! :See line 3370 ***
5 IF minutes<0 THEN minutes=minutes+60
time_elapsed=60*minutes+ end_time$(4 TO 5)-time_now$(4 TO 5)
S
268 cle=0
2745 IF new_tables_list(0)=1 THEN
2746 LET round$=" Round"
2747 ELSE
2748 LET round$=" rounds"
2749 END IF
2750 IF new_tables_list(0)>1 THEN
2751 intro "There will be "&new_tables_list(0)&round$
2752 END IF
2770 IF new_tables_list(0)>1 THEN intro "Round "&round
2780 PAPER 6:INK 0
3185 IF minutes<0 THEN minutes=minutes+60
3190 time_elapsed=60*minutes+ end_time$(4 TO 5)- time_now$(4 TO 5)
3250 PRINT
3270 LET cycle=cycle+1
3275 IF cycle>1 OR round>1 THEN
3276 CLS
3280 AT 5,4:PRINT "You got ";correct total; " out of ";cycle*round*10;" correct.
3294 LET average_time_taken=INT(total_time/(cycle*round)+.5)
3295 AT 9,4:PRINT "You took ";average_time_taken;" seconds per round."
3301 INK 2:FLASH 1
3302 IF average_time_taken>60 THEN
3303 AT 11,4:PRINT "Good try ";name$;"!":PAUSE 150
3304 AT 11,4:PRINT "Why not have another go?"
3305 ELSE
3306 AT 11,4:PRINT "Well done ";name$;"!"
3307 END IF
3308 PAUSE 150:INK 0:FLASH 0
3309 END IF
4551 IF level$(1) INSTR "TQ" THEN
4552 INK 2:AT 13,2: PRINT "Press <ENTER> if your name is not"
4553 AT 14,INT((40-LEN(name$))/2):PRINT name$:INK 0
4554 END IF
4570 AT 16,4:PRINT "Press R to Repeat"
4580 AT 17,4:PRINT "Press M to obtain Menu"
4600 AT 18,4:PRINT "Press ";change$(1);" to move on to a ";change$
4630 AT 19,4: INPUT "Which?"; option$
4635 IF option$="" AND level$(1) INSTR "TQ" THEN RUN

```

I was pleased to see my *Multiplication Tables* program in the February issue. There is a revised version which removes a few screen blinks and implements the multi-user option. I have also cured the negative time-clapsed bug which arises when readings of Date\$ straddle an hour.

If readers type-in the following listing and merge it with that published in the February issue they will have the updated version. Also, deleting the Mode 8: from line 610 and replacing the text+line parameter of 18 for the rub\_out procedure by 19-INT((LEN(option\$)+10)/37) in lines 4680 and 4710 will remove a few more blinks and tidy the screen when the user fills the keyboard buffer.

**Ron Allpress,  
Thwaite,  
Suffolk.**

```

1 CLEAR
2 MODE 8:CSIZE 1,1:PAPER 2:INK 7:CLS
3 AT 2,INT((28-LEN("LOADING"))/2):PRINT "LOADING"
4 AT 4,INT((28-LEN("Multiplication Tables"))/2):
PRINT "Multiplication Tables"
5 PAUSE 200:CSIZE 0,0:PAPER 1:INK 6:CLS
6 LRUN mdv1_Multiplication_tables

```



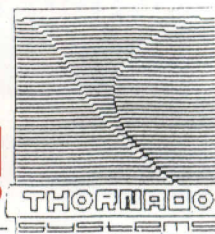


# QL ALLIANCE SUB



Telephone: 0388-450610, Fax: 0388-609845, Prestel MBX: 219998590.  
Telex: 934999 TXLINK G (Quoting Reference No. 219998590)

0388 450610 **THE QL'S** 0388 450610



# REVOLUTION HAS ARRIVED!

## QL SUB', THORNADO, QLCF, AND MORE- ALLIANCE OF QL USER GROUPS! —

- UK DISTRIBUTOR -

**FREE Magazine and Catalogue on Request.  
Just Ask for Information.**

FREE ON REQUEST  
**OL S.U.B.**

A collage of various computer-related documents and forms. In the foreground, a large white banner with bold black text reads "DUE TO THE RESPONSE OFFER EXTENDED TO 1st AUGUST". Behind the banner are several overlapping documents:

- SUB BUREAU**: A document with the title "SUB BUREAU" in large red letters, followed by "PROFESSIONAL COMPUTER SERVICES AND SUPPORT". Below it, there's a section titled "SPECIALIST" and another titled "LIST".
- REQUIREMENTS**: A document with the word "REQUIREMENTS" at the top.
- Form with fields**: A form with several fields, some containing handwritten or typed text like "the best price", "the other side of the bank", "the company", "the place", "the time", "the date", "the name", "the address", "the phone number", "the fax number", "the email address", "the website", "the social media", "the contact information", "the terms and conditions", "the privacy policy", "the disclaimer", "the liability", "the indemnification", "the arbitration", "the jurisdiction", "the governing law", "the applicable law", "the relevant law", "the local law", "the national law", "the international law", "the public law", "the private law", "the civil law", "the criminal law", "the administrative law", "the constitutional law", "the judicial law", "the executive law", "the legislative law", "the regulatory law", "the statutory law", "the common law", "the customary law", "the religious law", "the moral law", "the ethical law", "the professional law", "the academic law", "the scientific law", "the artistic law", "the literary law", "the musical law", "the theatrical law", "the cinematic law", "the sporting law", "the gaming law", "the gambling law", "the betting law", "the lottery law", "the casino law", "the poker law", "the roulette law", "the blackjack law", "the craps law", "the dice law", "the cards law", "the slots law", "the keno law", "the bingo law", "the scratch-off law", "the instant win law", "the pull-tab law", "the ticket-in machine law", "the coin-operated law", "the pay-per-play law", "the pay-per-line law", "the pay-per-spin law", "the pay-per-hand law", "the pay-per-roll law", "the pay-per-draw law", "the pay-per-bet law", "the pay-per-wager law", "the pay-per- wagering law", "the pay-per-gaming law", "the pay-per-recreation law", "the pay-per-leisure law", "the pay-per-hobby law", "the pay-per-pastime law", "the pay-per-amusement law", "the pay-per-entertainment law", "the pay-per-recreation law", "the pay-per-leisure law", "the pay-per-hobby law", "the pay-per-pastime law", "the pay-per-amusement law", "the pay-per-entertainment law".

**B. USER BUREAU**

PROFESSIONAL COMPUTER SERVICES AND SUPPLIES

NAME: A. SAMPLE

SUBSCRIPTION No: M040969QL

TEL: 0388-450610, FAX: 0388-609845  
TELEX: 934999 TXLINK G, REF: 219998590  
PRESTEL MBX: 219998590

PO BOX 3, SHILDON, DL4 2LW

## SUBSCRIPTION OFFER!

## What do we offer?

A light at the end of the tunnel... A full-time telephone help-line is backed up by written support, and a wide range of special services developed only for our readers. 12 issues of "QL SUB" magazine at (we hope) monthly intervals. Savings on hardware/software. Unbiased and informed advice.

Richard J. Turner.

Richard J. Turner.  
(Editor, "QL SUB")

## Who can benefit?

All sorts of people read "QL SUB"  
in all sorts of places- Papua New  
Guineau, Iran AND Iraq, Australia,  
United States, India, France, Oman,  
Zimbabwe, Libya, Greece, Poland...

**£15 (UK ONLY)**  
USUAL RATE  
STARTS £20

£20 Europe, £25 World.  
OFFER EXTENDED TO 1st AUG.

PHONE US NOW  
FOR BEST PRICES

Auckland Business Centre, St. Helen's Auckland,  
BISHOP AUCKLAND, Co. Durham, ENGLAND, DL14 9TX.

## FRIENDLY ADVICE ON COMPUTERS

**PO Box 3, Shildon<sup>(UK)</sup>, DL4 2LW**

Please Note- We use PO BOX 3 for convenience. If you would prefer to write to, or order from our business premises direct you are welcome to do so. Of course, you can call too, but please arrange a time to call.

WE WELCOME CALLERS BY APPOINTMENT ONLY!

QL Bulletin Board always on 0388-773737  
Running Every Day. All Day. Direct Dial!